Chapter 1
The Open Problem

Sherri Rose, Mark J. van der Laan

The debate over hormone replacement therapy (HRT) has been one of the biggest health discussions in recent history. Professional groups and nonprofits, such as the American College of Physicians and the American Heart Association, gave HRT their stamp of approval 15 years ago. Studies indicated that HRT was protective against osteoporosis and heart disease. HRT became big business, with millions upon millions of prescriptions filled each year. However, in 1998, the Heart and Estrogen-Progestin Replacement Study demonstrated increased risk of heart attack among women with heart disease taking HRT, and in 2002 the Women's Health Initiative showed increased risk for breast cancer, heart disease, and stroke, among other ailments, for women on HRT. Why were there inconsistencies in the study results?

Mammography gained relatively widespread acceptance as an effective tool for breast cancer screening in the 1980s. While there was still debate, several studies, including the Health Insurance Plan trial and the Swedish Two-County trial, demonstrated that mammography saved lives. This outweighed the minimal evidence against mammography. Thus, in 2009, many medical practitioners and nonprofits were surprised by the new recommendations from the U.S. Preventive Services Task Force. Among women without a family history, mammography was now only recommended for women aged 50 to 74. The previous guidelines started at age 40. Why was there a seemingly sudden paradigm shift?

A political scientist examines the effect of butterfly ballots in an election, which may in turn change local election laws. A group of economists studies the effect of microlending on the local economy in rural areas of Africa in hopes of promoting greater adoption of this practice. Public health policy decisions regarding how frequently to perform gynecological exams await the completion of several new investigations. The question then becomes, how does one translate the results from these studies, how do we take the information in the data, and draw effective conclusions?

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_1,

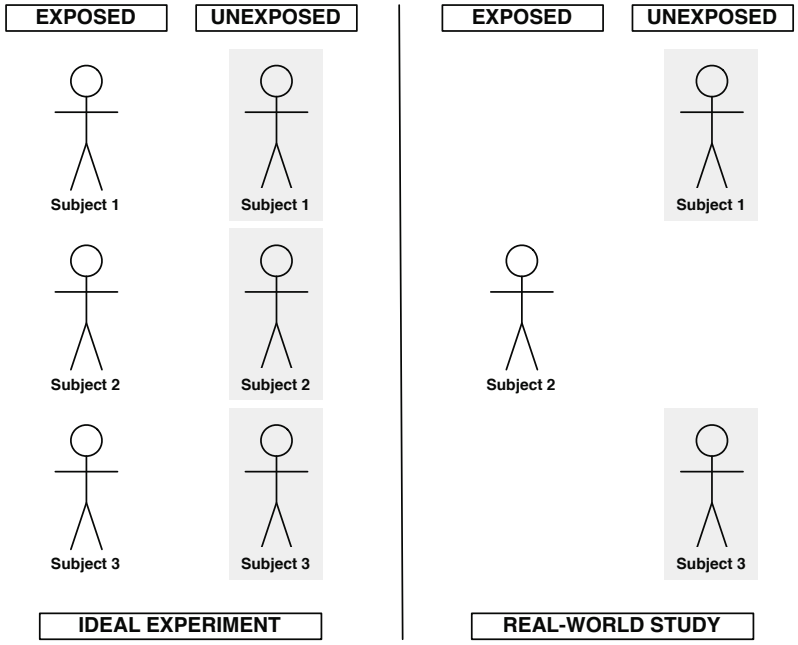

Fig. 1.1 Illustration of the "ideal experiment" vs. studies conducted in the real world

1.1 Learning from Data

One of the great open problems across many diverse fields of research has been obtaining causal effects from data. Data are typically sampled from a population of interest since collecting data on the entire population is not feasible. Frequently, the researcher is not interested in merely studying association or correlation in this sample data; she wants to know whether a treatment or exposure causes the outcome in the population of interest. If one can show that the treatment or exposure causes the outcome, we can then impact the outcome by intervening on the treatment or exposure.

Just what type of studies are we conducting? The often quoted "ideal experiment" is one that cannot be conducted in real life. Let us say we are interested in studying the causal effect of a toxin on death from cancer within 5 years. In an ideal experiment, we intervene and set the exposure to *exposed* for each subject. All subjects are followed for 5 years, where the outcome under this exposure is recorded. We then go back in time to the beginning of the study, intervene, and set all subjects to *not exposed* and *follow them under identical conditions* until the end of the study, recording the outcome under this exposure. As noted, we obviously cannot administer such a study since it is not possible to go back in time.

However, let's assume in principle there is a system where this ideal experiment could have been conducted. This experiment generates random variables. Say the experiment is that we sample a subject (i.e., draw a random variable) from a population and take several measurements on this subject. This experiment is repeated multiple times until we have sampled an a priori specified number (representing the sample size) of subjects. These random variables also have a true underlying probability distribution. Our observed data are realizations of these random variables. If we were to conduct our repeated experiment again, we would observe different realizations of these random variables.

Any knowledge we have about how these observed data were generated is referred to as a model. For example, it might be known that the data consist of observations on a number of independent and identically distributed (i.i.d.) random variables. What does i.i.d. mean? We are repeatedly drawing random variables from the same probability distribution, but each draw is mutually independent from all others. A common toy example used in statistics texts is the roll of a fair die. Say the experiment is to roll a die. We perform this experiment six times, each time rolling it following the same procedure (e.g., shaking). If we roll the die six times, we will see one set of realizations of these random variables, e.g., we observe a 1, 5, 3, 6, 2, and then a 5. Each roll of the die (i.e., each experiment) is independent from the previous roll. The observed unit in many cases may be the individual, where we sample repeatedly individual subjects from a population of interest. However, the observed unit can also be a household of individuals or a community of people.

So, our data are i.i.d. random variables, but the probability distribution of the random variable is typically completely unknown. This is also information we incorporate into our model. We will refer to this as a nonparametric model for the probability distribution of the random variable. (Do note, however, that assuming the data vector is i.i.d. in our nonparametric model is a real assumption, although one we will always make in this book.) Our model should always reflect true knowledge about the probability distribution of the data, which may often be a nonparametric model, or a semiparametric model that makes some additional assumptions. For example, perhaps it is known that the probability of death is monotonically increasing in the levels of exposure, and we want to include this information in our model.

The knowledge we have discussed thus far regarding our model pertains to our observed data and what we call the statistical model. The statistical model is, formally, the collection of possible probability distributions. The model may also contain extra information in addition to the knowledge contained in the statistical model. Now we want to relate our observed data to a causal model. We can do this with additional assumptions, and we refer to a statistical model augmented with these additional causal assumptions as the model for the observed data. These additional assumptions allow us to define the system where this ideal experiment could have been conducted. We can describe the generation of the data with nonparametric structural equations, intervene on treatment or exposure and set those values to *exposed* and *not exposed*, and then see what the (counterfactual) outcomes would

have been under both exposures. This underlying causal model allows us to define a causal effect of treatment or exposure.

One now needs to specify the relation between the observed data on a unit and the full data generated in the causal model. For example, one might assume that the observed data corresponds with observing all the variables generated by the system of structural equations that make up the causal model, up till background factors that enter as error terms in the underlying structural equations. The specification of the relation between the observed data and this underlying causal model allows one now to assess if the causal effect of interest can be identified from the probability distribution of the observed data. If that is not possible, then we state that the desired causal effect is not identifiable. If, on the other hand, our causal assumptions allow us to write the causal effect as a particular feature of the probability distribution of the observed data, then we have identified a target parameter of the probability distribution of the observed data that can be interpreted as a causal effect.

Let's assume that the causal effect is identifiable from the observed data. Our parameter of interest, here the causal effect of a toxin on death from cancer within 5 years, is now a parameter of our true probability distribution of the observed data. This definition as a parameter of the probability distribution of the observed data does not rely on the causal assumptions coded by the underlying causal model describing the ideal experiment for generating the desired full data, and the link between the observed data and the full data. Thus, if we ultimately do not believe these causal assumptions, the parameter is still an interesting statistical parameter. Our next goal becomes estimating this parameter of interest.

The open problem addressed in this book is the estimation of interesting parameters of the probability distribution of the data. This need not only be (causal) effect measures. Another problem researchers are frequently faced with is the generation of functions for the prediction of outcomes. For these problems, we do not make causal assumptions, but still define our realistic nonparametric or semiparametric statistical model based on actual knowledge. We view effect and prediction parameters of interest as features of the probability distribution of our data, well defined for each probability distribution in the nonparametric or semiparametric model. Statistical learning from data is concerned with efficient and unbiased estimation of these features and with an assessment of uncertainty of the estimator. Traditional approaches to estimation differ from this philosophy.

1.2 Traditional Approach to Estimation

We can sometimes implement one element of the ideal experiment: assigning a value for treatment or exposure in a controlled experiment. Controlled experiments are exactly what they sound like: they allow the investigator to control certain variables in the study. Randomized controlled trials (RCTs) are one type of controlled experiment where subjects are randomized to receive a specific level of treatment. For example, if each subject was assigned to one of two levels of treatment based on

the flip of a fair coin, the differences between the two groups would be solely due to treatment as all other factors would be balanced, up to random error. However, most studies are so-called observational studies where exposure or treatment is not assigned. In many cases it may not be ethical to set the exposure of interest in an RCT, or an RCT is cost prohibitive.

1.2.1 Experimental Studies

The randomization in RCTs suggests that we can estimate the causal effect of the treatment. For example, the difference of means between the treatment and control groups equals an additive causal effect. Indeed, this randomization of treatment in RCTs allows us to go from the observed data to the causal effect of interest. The difference in means can be estimated using a saturated regression of the outcome on treatment in a parametric statistical model where covariates are ignored. Since the regression is saturated (i.e., there is a parameter for each of the two observed values of treatment), this parametric statistical model is not making any unreasonable assumptions, and is thus actually nonparametric. Therefore, this parametric statistical model is not wrong, although the resulting estimator of the causal effect of the treatment is not the most efficient estimator. This so-called unadjusted estimator of the treatment effect is a nonparametric maximum likelihood estimator based on the reduced observations that only consist of the outcome and the treatment.

Suppose randomization did not occur perfectly due to chance (as is common), and there is a single covariate that is predictive of the outcome. We now have more subjects in the treatment or control group with a covariate that is predictive of the outcome, and this saturated regression ignoring the covariate will potentially contain a lot of residual error due to the exclusion of the covariate. Now, one might propose conditioning on the covariate and taking the difference in means for each stratum of the covariate. This results in a treatment effect within each stratum of the covariate. One might now estimate the causal effect of treatment as the average over all strata of these strata-specific treatment effects. This adjusted estimator of the treatment effect is a nonparametric maximum likelihood estimator based on the reduced observations that consist of the outcome, treatment, and this single covariate. This approach is generally still not efficient, since it only uses one of the measured covariates, but it is more efficient than the unadjusted treatment effect estimator. However, this strategy is not practical with multiple covariates, or even one continuous covariate, and starts to suffer in practical performance due to strata with a very small number of subjects.

So why not run regressions in parametric statistical models (incorporating all covariates) for RCTs? The short answer is simple: the Food and Drug Administration (FDA) does not allow it. We will explain why this is so in a few sections. For now it is sufficient to know that the FDA requires researchers to specify a priori the method of estimation, and it must rely on a statistical model that reflects true knowledge.

1.2.2 Observational Studies

Recall that observational studies do not involve randomization to treatment or exposure. In most observational studies, standard practice for effect estimation involves assuming a parametric statistical model and using maximum likelihood estimation to estimate the parameters in that statistical model. Let us be very clear again about what a statistical model is: the statistical model represents the set of possible probability distributions of the data.

In traditional practice, one assumes the actual data as observed in practice can be represented as observations of n i.i.d. random variables, and that the goal of the traditional modeling approach is to learn the true underlying probability distribution that generated the data. (This is different than the goal of causal effect estimation.) Maximum likelihood estimation uses the likelihood function to estimate the unknown parameter(s) in the statistical model. Solutions are often found by differentiating the log-likelihood with respect to these parameters, setting the resulting equation equal to zero, and solving. If the score equation has multiple solutions, the solution with the largest likelihood is selected.

This procedure is detailed in most introductory statistics books, although the pervasiveness of statistical software allows the user to implement maximum likelihood estimation without the need to understand these concepts. This also means the assumptions that come with the use of parametric statistical models are frequently not well understood or ignored.

We already acknowledged in Sect. 1.1 that we usually know very little about how our data were generated; thus the use of parametric statistical models is troublesome. We typically know that our data can be represented as a number (representing the sample size) of i.i.d. observations, which is an assumption in parametric statistical models, but we do not know the underlying probability distribution that generated the data. Parametric statistical models assume the underlying probability distribution that generated the data is known up to a finite number of parameters. It is an accepted fact within the statistical community that nonsaturated *parametric statistical models are wrong*. Thus, making an assumption known to be untrue is not the best approach. When this assumption is violated and the statistical model is misspecified, the estimate of the probability distribution can be extremely biased, and it is not even clear what the parameter estimates are even estimating. The bias resulting from statistical model misspecification cannot be overcome with a large sample size.

This brings us to another problem that arises when using misspecified parametric statistical models. The target parameter is not defined as a parameter of the true probability distribution for any possible probability distribution. The target parameter, when defined as a coefficient in a (misspecified) parametric statistical model, is only defined within that parametric statistical model, as if the statistical model were true. There is only correct inference if the parametric statistical model is correct, but we know it is wrong.

Lastly, the traditional approach does not make any explicit (untestable) causal assumptions linking the observed data to a system that generated the data. Thus, there

is no framework to make causal inference. There are also other assumptions that are part of the statistical model that are typically not addressed, such as positivity (discussed in Chaps. 2 and 10). When this (testable) assumption is violated, you may see groups of individuals where there is no experimentation in the treatment. For example, all the highly educated women received HRT, or all the wealthy women received mammograms. Since there are strata of certain covariates (e.g., level of education, socioeconomic status) where all subjects are treated, the regression will extrapolate what would have happened to these subjects had they not been treated, and this extrapolation is not based on any observed information.

To summarize, the use of parametric statistical models in observational studies is troublesome for several main reasons.

1. The statistical models are always misspecified in practice since we do not know the underlying data-generating distribution and we handle complex problems with many covariates.
2. The target parameter is not defined as a parameter of the true probability distribution that generated the data.
3. The traditional approach does not typically make causal assumptions allowing us to define the desired causal effect, and often neglects other key assumptions, such as the positivity assumption, that are part of the statistical model.

1.2.3 Regression in (Misspecified) Parametric Statistical Models

In this section we discuss briefly the traditional approach to effect estimation. Let us introduce our random variable O, which has probability distribution P_0. This is written $O \sim P_0$. Recall that a probability distribution P_0 assigns a probability to any possible event or set of possible outcomes for O. In particular, $P_0(O = o)$ for a particular value o of O can be defined as a probability if O is a discrete random variable, or we can use the concept of probability density if O is continuous. For simplicity and sake of presentation, we will often treat O as discrete so that we can refer to $P_0(O = o)$ as a probability.

We observe our random variable O n times, by repeating the same experiment n times. For a simple example, suppose our data structure is $O = (W, A, Y) \sim P_0$. We have a covariate or vector of covariates W, an exposure or treatment A, and a continuous outcome Y. These variables comprise the random variable O, which we observe repeatedly, and O has probability distribution P_0. Thus, for each possible value (w, a, y), $P_0(w, a, y)$ denotes the probability that (W, A, Y) equals (w, a, y). For example, the random variables $O_1, \ldots, O_n$ might be the result of randomly sampling n subjects from a population of patients, collecting baseline characteristics

W, assigning treatment or exposure A, and following the patients and measuring continuous outcome Y.

Suppose one poses a particular regression in a parametric statistical model, a so-called linear regression for the conditional mean of Y given A or Y given A and W. However, we leave the distributions of A and W unspecified. Linear regression in a parametric statistical model has varying levels of complexity, and what variables one includes impacts this complexity. The saturated regression for RCTs discussed in Sect. 1.2.1 includes only a treatment variable A. (This is sometimes called a crude regression.) For example, with a continuous outcome Y and a binary treatment A the regression of the conditional mean of Y given A is

$$E_0(Y \mid A) = \alpha_0 + \alpha_1 A.$$

The parameter $E_0(Y \mid A)$ is the conditional mean of Y given A, and (α_0, α_1) are the unknown regression parameters in the parametric statistical model for the conditional mean. We are estimating the regression $E_0(Y \mid A)$ based on the data $(A_1, Y_1), \ldots, (A_n, Y_n)$, ignoring the covariates W_i for subject i. Fitting this regression to the data will result in an estimate of the effect of treatment given by $\alpha_1 = E_0(Y \mid A = 1) - E_0(Y \mid A = 0)$.

In the analysis of observational studies, it is commonplace to include covariates associated with both A and Y in the regression, in an attempt to eliminate the contribution of these variables and isolate the effect of A on Y. With one covariate W, an example of such a regression in a parametric statistical model is

$$E_0(Y \mid A, W) = \alpha_0 + \alpha_1 A + \alpha_2 W.$$

The effect of A is again given by α_1, but α_1 now represents an effect of A adjusting for W, and is thus a different parameter of interest than the effect of A above. If effect modification is suspected, an interaction term between the effect modifier and A might be included:

$$E_0(Y \mid A, W) = \alpha_0 + \alpha_1 A + \alpha_2 W + \alpha_3 A \times W. \tag{1.1}$$

Effect modification between A and W occurs when the effect of A differs within strata of W. The consequence of including an interaction term in the regression is that there is now not one summary measure of the effect of A. For every level of W there is a different effect measure of A. For example, in the simple case where W is binary, such as smoking status, there will be two effect measures for A. If $W = 1$ indicates current smoker, the effect of A among current smokers is $\alpha_1 + \alpha_3$. When $W = 0$, α_3 is equal to zero thus the effect of A among current nonsmokers is α_1. As we add covariates and interaction terms to our regression, α_1 does not estimate a marginal population-level effect. In fact, each time we add a covariate or interaction the interpretation of the coefficients in the parametric statistical model changes.

In the situation where we only have one binary covariate, the regression specified in Eq. (1.1) is a saturated parametric statistical model. Let us also suppose the collection of the single covariate represents the truth, and there are no other covari-

ates that should have been measured. This parametric statistical model is therefore suitable in that it is *not misspecified*. However, we still want a marginal effect estimate of treatment. This marginal-effect could be defined as $\alpha_1 + \alpha_3 E_0(W)$, where $E_0(W)$ denotes the true marginal mean of W. A simple nonparametric maximum likelihood estimator will accomplish this for the simple case posed here. But what happens when you have a continuous covariate? Or we have an increasing number of covariates? This approach to fitting a saturated linear regression quickly becomes problematic since the number of coefficients will grow exponentially with the number of covariates.

High-dimensional data have become increasingly common, and researchers often have dozens, hundreds, or even thousands of potential covariates to include in their parametric statistical model. Not only does this provide an impossible challenge to correctly specify the parametric statistical model for the conditional mean, but the complexity of the parametric statistical model may also increase to the point that there are more unknown parameters than observations. A fully saturated parametric statistical model will usually result in a gross overfit of the data. In addition, the true functional, $(A, W) \rightarrow E_0(Y \mid A, W)$, mapping the treatment and covariates into the conditional mean, might be described by a complex function not easily approximated by main terms or simple two-way interactions.

1.2.4 The Complications of Human Art in Statistics

We now highlight further the innate challenges of parametric statistical models and the problematic human art component of data analysis. Returning to our toxin and cancer study from Sect. 1.1, where an indicator of death is the outcome, let's say that the principal investigator (PI) asserts smoking status is the only relevant covariate that we must control for in our analysis. The PI also says to use the following logistic linear regression in a parametric statistical model for the probability of death, where the α_is are the unknown regression parameters in the statistical model:

$$P_0(Y = 1 \mid A, W) = \text{expit}(\alpha_0 + \alpha_1 A + \alpha_2 W).$$

Another subject matter expert on the project enters the conversation and says that one must also control for age and gender. Smoking is now denoted W_1, with age as W_2 and gender W_3. The covariates can be represented as a vector $W = \{W_1, W_2, W_3\}$ and the logistic linear regression given by

$$P_0(Y = 1 \mid A, W) = \text{expit}(\alpha_0 + \alpha_1 A + \alpha_2 W_1 + \alpha_3 W_2 + \alpha_4 W_3).$$

A data analyst enters the picture and explains that all covariates measured at baseline, listed in Table 1.1, should be thrown into a logistic linear regression. Using the results of this regression fit, all W_is with coefficients that do not have a p-value smaller than 0.05 should be removed from the list. The regression should then be fit again in a new (different) parametric statistical model with the variables remaining

Table 1.1 Baseline covariates from a study examining the effect of a toxin on death from cancer

W_i	Covariate
W_1	Smoking status
W_2	Age
W_3	Gender
W_4	Health status
W_5	Cardiac event
W_6	Chronic illness

in the list. This continues until all coefficients in front of the W_is in the regression have a p-value of less than 0.05. The regression coefficient α_1 in front of A changes with each new regression. It is highly dependent on which variables are included.

One can quickly see, even in this simplified example, the impossible challenge involved in selecting which variables to include in the parametric statistical model, and thereby assigning the underlying probability distribution of Y, conditional on the treatment and covariates, that generated the data up to a finite number of unknown parameters. The problem that we stress again here is that we do not know the true probability distribution of the data up to a finite number of unknown parameters.

The inference made using parametric statistical models assumes that the parametric statistical model is correct and was a priori selected. If the parametric statistical model is wrong, our estimates will approximate a noninterpretable parameter, and thereby be biased for the true hypothesized target parameter one had in mind under the assumption that the parametric statistical model was true. If we run several models with the full data, the statistical inference (e.g., the p-values) is meaningless, and this statistical model should be selected before looking at the data to avoid bias.

In addition, if the parametric statistical model was not a priori specified but data-adaptively selected as the data analyst suggests, then the statistical inference is misleading, claiming a certainty that does not exist. The final parametric statistical model is reported as if it were the only one considered and evaluated. The data analyst has performed a procedure that began the moment the data were used. In other words, once you start using the data, your estimation method has also started. Therefore, our data analyst has selected an approach that, while very common, blatantly leaves us with faulty inference.

Even without the approach defined by the data analyst, the PI and the subject matter expert might run both of their regressions and then decide between them based on the results. It should not be overlooked that the process of looking at the data, examining coefficient p-values, and trying multiple statistical models is not only incredibly prevalent but is taught to students learning statistics.

This is the human art component we eluded to in Sect. 1.2. The moment we use post-hoc arbitrary criteria and human judgment to select the parametric statistical model after looking at the data, the analysis becomes prone to additional bias. This bias manifests in both the effect estimate and the assessment of uncertainty for that estimator (i.e., standard errors). One cannot even define the procedure that was used

as a function of the data so that more appropriate standard errors can be calculated (e.g., by use of bootstrapping). Statistics is not an art, it is a science.

Standard practice focuses on estimating $E_0(Y \mid A, W)$ with an assumed parametric statistical model. One then extracts the coefficient in front of A as the effect estimate, ignoring that we know that most *parametric statistical models are wrong.* This criticism extends in general to estimation procedures (e.g., prediction) using misspecified parametric regression models. There is a more natural way to think about our parameter of interest, which we introduced abstractly in Sect. 1.1. The definitions of the data, model, and parameter will allow us to target parameters that are frequently of interest, such as causal effects. These concepts will be developed more concretely in the next section, and additionally in Chap. 2, as we set aside the traditional approach to effect estimation.

1.3 Data, Model, and Target Parameter

Our discussion of the data, model, and target parameter has been relatively abstract up to this point. We formalize these concepts in this section using notation. We define O as the random variable with P_0 as the corresponding probability distribution of interest. We write $O \sim P_0$ to mean that the probability distribution of O is P_0. Our random variable, which we observe n times, could be defined in a simple case as $O = (W, A, Y) \sim P_0$ if we are without common issues such as missingness and censoring. W, A, and Y are as defined in Sect. 1.2.3.

Complex data structures. While the data structure $O = (W, A, Y) \sim P_0$ makes for effective examples, data structures found in practice are frequently more complicated. Suppose we have a right-censored data structure. Right censoring means that we do not observe a particular variable or variables to the end of the study or time period. For example, if we are following subjects for 5 years, some subjects may drop out of the study for various reasons (e.g., relocation, death, voluntarily ending participation). If we are planning to measure an outcome Y, such as developing liver cancer, within 5 years of baseline, those subjects that drop out before 5 years (i.e.,

are censored) will not have measurements across the whole time period. All subjects will be censored at year 5 if they have not already been censored, but subjects that are observed for the full 5 years provide us with the desired full-data structure and are thereby referred to as uncensored.

Censoring is always defined with respect to a desired full-data structure. This type of censoring of a desired full-data structure is referred to as right censoring since timelines are frequently numbered from left to right, and it is some portion of the right side that is censored. Now, for each subject we will observe their time of censoring, and we may observe their time to event. For example, subject 1 may develop liver cancer at year 3. Another subject may be censored at year 2 due to dropout, and we never observe whether they develop liver cancer within the 5 years. Thus our data structure now has added complexity. We have T representing time to event Y, C a censoring time, $\tilde{T} = \min(T, C)$ which represents the T or C that was observed first, and $\Delta = I(T \leq \tilde{T}) = I(C \geq T)$ an indicator that T was observed at or before C. We then define $O = (W, A, \tilde{T}, \Delta) \sim P_0$. This is another example of a possible data structure.

1.3.1 The Model

We are considering the general case that one observed n i.i.d. copies of a random variable O with probability distribution P_0. The data-generating distribution P_0 is also known to be an element of a statistical model $\mathcal{M}$, which we write $P_0 \in \mathcal{M}$. Formally, a statistical model $\mathcal{M}$ is the set of possible probability distributions for P_0; it is a collection of probability distributions. What if all we know is that we have n i.i.d. copies of O? Well, then we've stated what we know, thus this can be our statistical model, which we call a nonparametric statistical model. We don't need to assign a parametric form to the distribution of our data; it is simply known to be an element of a nonparametric statistical model $\mathcal{M}$.

We might also consider a semiparametric statistical model if we have additional information about the way our data were generated that puts restrictions on the data-generating distribution P_0. For example, we may know that the effect of exposure A on the mean outcome is linear. Note, though, that semiparametric statistical models can be wrong by not containing the true P_0 if our "knowledge" is faulty. While we might have additional knowledge, we do not have enough knowledge to parameterize P_0 by a finite-dimensional parameter. These nonparametric and semiparametric statistical models should represent true knowledge about the underlying mechanism generating the data, that is, they are supposed to contain the true probability distribution P_0 of the experimental data.

We will frequently use *semiparametric* to include both nonparametric and semiparametric, such as the phrase "semiparametric estimation" referring to estimation in a nonparametric or semiparametric statistical model. When semiparametric excludes nonparametric and we make additional assumptions, this will be explicit.

Statistical model vs. model. A statistical model can be augmented with additional (causal) assumptions providing a parameterization so that $\mathcal{M} = \{P_\theta : \theta \in \Theta\}$, where the space of θ-values, Θ, is itself infinite dimensional. Even though such a parameterization does not change the statistical model, thereby providing nontestable causal assumptions, it does allow one to enrich the interpretation of $\Psi(P_0)$ in terms of a statement of an underlying truth θ_0. We refer to the statistical model augmented with a parameterization as a model. We will return to the issue of modeling, thereby making (causal) assumptions that go beyond specifying a statistical model $\mathcal{M}$, in Chap. 2. The important take-home message for now is that the statistical model is the only relevant information for the estimation problem, while the additional (causal) assumptions will provide enriched (or misleading, if wrong) interpretations of the target parameter.

1.3.2 The Target Parameter

What are we trying to learn from our data? Often the question of interest is related to quantifying some difference in the probability distribution of an outcome of interest between the treated and untreated or the exposed and unexposed groups. We want to understand the effect of treatment or exposure on the probability distribution of the outcome of interest. This difference could be measured on an additive scale or multiplicative scale, such as a relative risk or odds ratio.

Either way, once an agreement is reached concerning what one wants to learn, we can explicitly define the target parameter of the probability distribution P_0 as some function of P_0: $\Psi(P_0)$ for some function Ψ that maps the probability distribution P_0 into the target feature. That is, we are interested in estimating a parameter $\Psi(P_0)$ of the probability distribution $P_0 \in \mathcal{M}$, which is known to be an element of a nonparamteric or semiparametric statistical model $\mathcal{M}$. The parameter $\Psi(P_0)$ is a function of the unknown probability distribution P_0. We are not interested in estimating an effect defined by a coefficient of a (misspecified) parametric statistical model. Rather, we define a parameter as a feature of the true probability distribution P_0 of the data using true knowledge we have about P_0 as embodied by the statistical model $\mathcal{M}$. Thus, we are explicitly confronted with the fact that we need to know how to define our target parameter as a feature of P_0: it does not suffice to grab a parametric statistical model and just target the coefficients in that model.

First, one needs to define the parameter of interest as a function of the data-generating distribution varying over the nonparametric or semiparametric statistical model. Many practitioners are used to thinking of their parameter in terms of a regression coefficient, but that is often not possible in realistic nonparametric and semiparametric statistical models. Instead, one has to carefully think about what feature of the distribution of the data one wishes to target. With an experimental unit-specific data structure $O = (W, A, Y) \sim P_0$, the risk difference is the following function of the distribution P_0 of O:

$$\Psi(P_0) = E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)],$$

where $E_0(Y \mid A = a, W)$ is the conditional mean of Y given $A = a$ and W. Here A is binary and therefore a takes on two values, 1 and 0. E_W indicates that we take the average over the observed distribution of our covariate(s) W. Uppercase letters represent random variables and lowercase letters are a specific value for that variable. For example, if all variables are discrete, $P_0(W = w, A = a, Y = y)$ assigns a probability to any possible outcome (w, a, y) for $O = (W, A, Y)$. P_0 is like a calculator: we input (w, a, y) and it returns a probability. $\Psi(P_0)$ for the risk difference can then also be written:

$$\Psi(P_0) = \sum_w \Big[\sum_y y P_0(Y = y \mid A = 1, W = w) - \sum_y y P_0(Y = y \mid A = 0, W = w) \Big] P_0(W = w),$$

where

$$P_0(Y = y \mid A = a, W = w) = \frac{P_0(W = w, A = a, Y = y)}{\sum_y P_0(W = w, A = a, Y = y)}.$$

After obtaining an estimate of $\Psi(P_0)$ and a confidence interval, we can provide two interpretations, one as a purely statistical parameter of P_0, and one as a causal parameter under additional causal assumptions representing a causal model that goes beyond the specification of the statistical model $\mathcal{M}$. We discuss these causal assumptions in detail in Chap. 2.

1.3.3 Summary of Concepts

1. **Data.** Our data are comprised of n i.i.d. copies of a random variable $O \sim P_0$. P_0 is the true probability distribution for O.
2. **Model.** Our statistical model $\mathcal{M}$ is nonparametric or semiparametric and represents only what we know about our data-generating distribution P_0. $\mathcal{M}$ is the set of possible probability distributions for P_0. Our model includes possible additional causal assumptions, allowing an enriched interpretation of the parameter of interest.
3. **Target parameter.** Our parameter $\Psi(P_0)$ is a particular feature of the unknown probability distribution P_0. The explicit definition of this mapping Ψ on the statistical model requires that one defines $\Psi(P)$ at each P in the statistical model. The parameter typically has two interpretations, one as a parameter $\Psi(P_0)$ of a probability distribution P_0 and one as a causal parameter under additional (causal) assumptions to be discussed in Chap. 2.

1.4 The Need for Targeted Estimators

Let us step back for a moment. Suppose you were handed ten textbooks and told you would be asked one question in 12 h. The question might require understanding portions of several of these books. However, you are not told what the question is going to be. How would you prepare for such a test? You do not have time to read all ten textbooks, let alone master the material contained within them. You might read the chapter abstracts from each book in order to learn basic summary information.

Now, suppose you were handed the same ten textbooks, but instead you were told the question you would be asked 12 h later. Would this change your approach to studying? Yes! Since you know what question will be asked, you can more carefully discard books that will be completely unnecessary, keeping only those books with relevant chapters. You are able to spend 12 h working through the pertinent chapters and then give a thoughtful precise answer on the one question test.

This theoretical situation has a direct parallel to nontargeted learning vs. targeted learning. Maximum likelihood estimation in misspecified parametric statistical models is nontargeted learning; one estimates all the parameters (coefficients) in a parametric statistical model. One uses an empirical criterion that is only concerned with the overall fit of the entire probability distribution of the data instead of only the parameter of interest; we are trying to master all the books, spreading error uniformly across all content, when we only care about very specific portions of each book. The overall fit of the probability distribution based on the data set is then used to evaluate the target parameter of the probability distribution, i.e., the question is answered with the nontargeted fit of the distribution of the data. For a small *true* parametric statistical model, containing the true probability distribution, one with few terms or few unknown coefficients, the performance of the maximum likelihood estimator of the target parameter with regard to mean squared error may be satisfactory. However, the bigger the statistical model, the more problematic nontargeted learning becomes. We have no problem with maximum likelihood estimation for relatively low-dimensional parametric statistical models if they are correct, but this is not the case in practice, and we wish for our statistical models to represent true knowledge. Indeed, in semiparametric statistical models, maximum likelihood estimation breaks down completely. With targeted learning, we focus on our known question of interest; we focus on the relevant information in the books, and rank the information by its relevance for the question of interest.

1.5 Road Map for Targeted Learning

The first six chapters of this textbook are meant to provide the reader with a firm grasp of the targeted learning road map and the solution to prediction and causal inference estimation problems: super learning and targeted maximum likelihood estimation (TMLE). For the sake of presentation, in these introductory chapters we will focus on the data structure $(W, A, Y) \sim P_0$, the nonparametric statistical

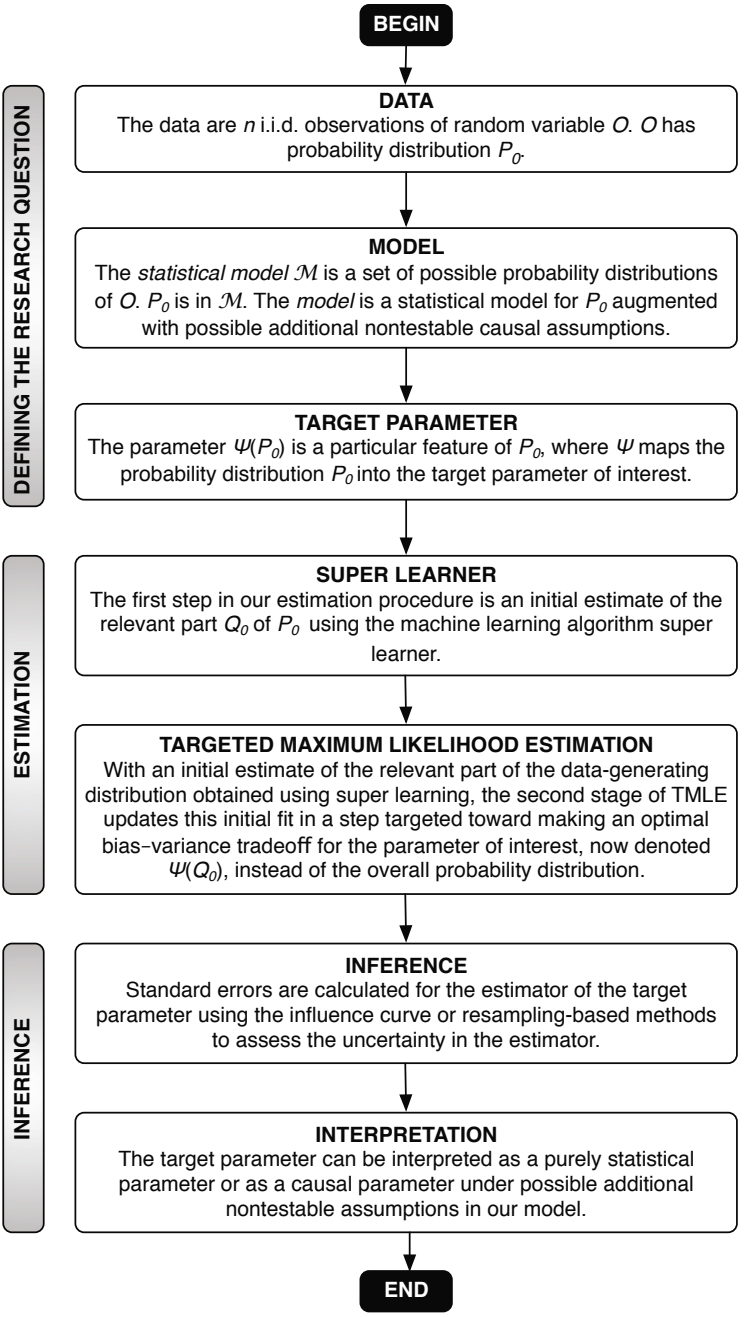

Fig. 1.2 Road map for targeted learning

model $\mathcal{M}$, and the additive causal effect target parameter $\Psi(P_0) = E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$. Our estimator of the treatment effect will be obtained by plugging in a (targeted) estimator of (or the relevant part of) P_0 into the parameter mapping Ψ. Such an estimator is called a plug-in or substitution estimator. Substitution estimators have the advantage of fully respecting the constraints implied by the statistical model $\mathcal{M}$ and respecting that the target parameter is a very specific function of P_0. As a consequence, substitution estimators are generally robust, even in small samples.

This first chapter was intended to motivate the need for improved estimation methods, highlight the troublesome nature of the traditional approach to estimation, and introduce important concepts such as the data, model, and target parameter. We develop the following concepts, as part of the road map for targeted learning, in the remaining five introductory chapters.

Defining the model and target parameter. By defining a structural causal model (SCM), we specify a model for underlying counterfactual outcome data, representing the data one would be able to generate in an ideal experiment. This is a translation of our knowledge about the data-generating process into causal assumptions. We can define our target parameter in our SCM, i.e., as a so-called causal effect of an intervention on a variable A on an outcome Y. The SCM also generates the observed data O, and one needs to determine if the target parameter can be identified from the distribution P_0 of O alone. In particular, one needs to determine what additional assumptions are needed in order to obtain such identifiability of the causal effect from the observed data.

Super learning for prediction. The first step in our estimation procedure is an initial estimate for the part of the data-generating distribution P_0 required to evaluate the target parameter. This estimator needs to recognize that P_0 is only known to be an element of a semiparametric statistical model. That is, we need estimators that are able to truly learn from data, allowing for flexible fits with increased amounts of information. We introduce cross-validation and machine learning as essential tools and then present the method of super learning for prediction with its theoretical grounding, demonstrating that super learning provides an optimal approach to estimation of P_0 (or infinite-dimensional parameters thereof) in semiparametric statistical models. Since prediction can be a research question of interest in itself, super learning for prediction is useful as a standalone tool as well.

TMLE. With an initial estimate of the relevant part of the data-generating distribution obtained using super learning, we are prepared to present the remainder of the TMLE procedure. The second stage of TMLE updates this initial fit in a step targeted towards making an optimal bias–variance tradeoff for the parameter of interest, instead of the overall probability distribution P_0. This results in a targeted estimator of the relevant part of P_0, and thereby in a corresponding substitution estimator of $\Psi(P_0)$.

Many of the topics we have presented in this road map may be new to you. They will be explained in detail in the coming chapters. This brief road map is introduced

for the reader to see where we are going, how the pieces fit together, and why we will present material in this order.

1.6 Notes and Further Reading

We motivated this chapter with two real-world debates: HRT and screening guidelines for breast cancer. In a *New York Times* piece, Taubes (2007) discussed the merits of epidemiology using the HRT studies as an example. For those interested in reading more about this topic, it is an excellent comprehensive starting point with thorough references. For the statistician and researcher, it also raises one of the questions we seek to answer with this text. Can we estimate causal effects from observational studies? Two starting points for the mammography debate include U.S. Preventive Services Task Force (2009) for the official recommendation statement on breast cancer screenings, as well as Freedman et al. (2004) for a qualitative review of breast cancer mammography studies.

For additional background on study designs and covariate adjustment we direct readers to Rothman and Greenland (1998) and Jewell (2004). For a readable introductory statistics text on traditional regression techniques and key statistics concepts such as the central limit theorem (CLT) we refer readers to Freedman (2005).

A popular article drawing attention to false research findings, due in part to current statistical practice, is Ioannidis (2006). Ioannidis was also interviewed in journalist David H. Freedman's new book, *Wrong: Why Experts Keep Failing Us—And How to Know When to Trust Them*. This text focuses on problems in research fields, including the way data are analyzed and presented to the public (Freedman 2010).

George Box famously discussed that (parametric) statistical models are wrong, but may be useful (Box and Draper 1987). As presented in this chapter, misspecified parametric statistical models may not perform terribly for low-dimensional data structures and small sample sizes. Over 20 years after Box's statements, data sets have become increasingly high dimensional, and large studies are very common. We are also still left with the issue that the coefficients in misspecified parametric statistical models do not represent the target parameter of interest. Therefore, the usefulness of misspecified parametric statistical models is extremely limited. Note, however, that maximum likelihood estimators according to candidate parametric working statistical models can be included in the library of the super learner, discussed in Chap. 3, and can play a useful role in that manner.

The use of data-adaptive tools can be beneficial, although we discuss in this chapter (Sect. 1.2.4) a commonly used data-adaptive procedure in parametric statistical models that provides faulty inference. Data-adaptive methods, when guided by a priori benchmarks in a nonparametric or semiparametric statistical model, are advantageous for prediction and discussed in detail in Chap. 3. We use the terms *data-adaptive* and *machine learning* interchangeably in this text. Targeted estimators will be discussed in Chaps. 4–6.

Chapter 2
Defining the Model and Parameter

Sherri Rose, Mark J. van der Laan

Targeted statistical learning from data is often concerned with the estimation of causal effects and an assessment of uncertainty for the estimator. In Chap. 1, we identified the road map we will follow to solve this estimation problem. Now, we formalize the concepts of the model and target parameter. We will introduce additional topics that may seem abstract. While we attempt to elucidate these abstractions with tangible examples, depending on your background, the material may be quite dense compared to other textbooks you have read. Do not get discouraged. Sometimes a second reading and careful notes are helpful and sufficient to illuminate these concepts. Researchers and students at UC Berkeley have also had great success discussing these topics in groups. If this is your assigned text for a course or workshop, meet outside of class with your fellow classmates. We guarantee you that the effort is worth it so you can move on to the next step in the targeted learning road map. Once you have a firm understanding of the core material in Chap. 2, you can begin the estimation steps.

This chapter is based on methods pioneered by Judea Pearl, and we consider his text *Causality*, recently published in a second edition (Pearl 2009), a companion book to our book. Causal inference requires both a causal model to define the causal effect as a target parameter of the distribution of the data *and* robust semiparametric efficient estimation, with his book covering the former and ours the latter. We start by succinctly summarizing the open problem:

> The statistical estimation problem begins by defining a statistical model $\mathcal{M}$ for P_0. The statistical model $\mathcal{M}$ is a collection of possible probability distributions P of O. P_0 is the true distribution of O. The estimation problem requires the description of a target parameter of P_0 one wishes to learn from the data. This definition of a target parameter requires specification of a mapping Ψ one can then apply to P_0. Clearly, this mapping Ψ needs to be defined on any possible probability distribution in the statistical model $\mathcal{M}$. Thus Ψ maps any $P \in \mathcal{M}$ into a vector of numbers $\Psi(P)$. We write the mapping as $\Psi : \mathcal{M} \to \mathbb{R}^d$ for a

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_2,

d-dimensional parameter. We introduce ψ_0 as the evaluation of $\Psi(P_0)$, i.e., the true value of our parameter. The statistical estimation problem is now to map the observed data $O_1, \ldots, O_n$ into an estimator of $\Psi(P_0)$ that incorporates the knowledge that $P_0 \in \mathcal{M}$, accompanied by an assessment of the uncertainty in the estimator.

In the following sections, we will define a model that goes beyond a statistical model by incorporating nontestable assumptions, define a parameter of interest in that model that can be interpreted as a causal effect, determine the assumptions to establish the identifiability of the causal parameter from the distribution of the observed data, and, finally, based on this modeling and identifiability exercise, commit to a statistical model (i.e., $\mathcal{M}$) and target parameter (i.e., Ψ).

Recall that the data $O_1, \ldots, O_n$ consist of n i.i.d. copies of a random variable O with probability distribution P_0. For a data structure, such as $O = (W, A, Y)$ with covariates W, exposure A, and outcome Y discrete, which we use as a simple example in this chapter, uppercase letters represent random variables and lowercase letters are a specific value for that variable. For example, if all variables are discrete, $P_0(W = w, A = a, Y = y)$ assigns a probability to any possible outcome (w, a, y) for $O = (W, A, Y)$.

2.1 Defining the Structural Causal Model

We first specify a set of endogenous variables $X = (X_j : j)$. Endogenous variables are those variables for which the structural causal model (SCM) will state that it is a (typically unknown) deterministic function of some of the other endogenous variables and an exogenous error. Typically, the endogenous variables X include the observables O, but might also include some nonobservables that are meaningful and important to the scientific question of interest. Perhaps there was a variable you did not measure, but would have liked to, and it plays a crucial role in defining the scientific question of interest. This variable would then be an unobserved endogenous variable. For example, if you are studying the effect of hepatitis B on liver cancer, you might also want to measure hepatitis C and aflatoxin exposure. However, suppose you know the role aflatoxin plays in the relationships between hepatitis B and liver cancer, but you were unable to measure it. Aflatoxin exposure is, therefore, an unobserved endogenous variable. Liver cancer, hepatitis B, and hepatitis C are observed endogenous variables.

In a very simple example, we might have $j = 1, \ldots, J$, where $J = 3$. Thus, $X = (X_1, X_2, X_3)$. We can rewrite X as $X = (W, A, Y)$ if we say $X_1 = W$, $X_2 = A$, and $X_3 = Y$. Let W represent the set of baseline covariates for a subject, A the treatment or exposure, and Y the outcome. All the variables in X are observed. Suppose we are interested in estimating the effect of leisure-time physical activity (LTPA) on mortality in an elderly population. A study is conducted to estimate this effect where

we sample individuals from the population of interest. The hypothesis is that LTPA at or above current recommended levels decreases mortality risk. Let us say that LTPA is a binary variable $A \in \{0, 1\}$ defined by the recommended level of energy expenditure. For all subjects meeting this level, $A = 1$ and all those below have $A = 0$. The mortality outcome is also binary $Y \in \{0, 1\}$ and defined as death within 5 years of the beginning of the study, with $Y = 1$ indicating death. W includes variables such as age, sex, and health history.

For each endogenous variable X_j one specifies the parents of X_j among X, denoted $Pa(X_j)$. In our mortality study example above, the parent of A is the set of baseline covariates W. Thus, $Pa(A) = W$. The specification of the parents might be known by the time ordering in which the X_j were collected over time: the parents of a variable collected at time t could be defined as the observed past at time t. This is true for our study of LTPA; $W = \{\text{age, sex, health history}\}$ all occur before the single measurement of LTPA. Likewise, LTPA was generated after the baseline covariates and before death but depends on the baseline covariates. Death was generated last and depends on both LTPA and the baseline covariates. We can see the time ordering involved in this process: the baseline covariates occurred before the exposure LTPA, which occurred before the outcome of death: $W \rightarrow A \rightarrow Y$.

We denote a collection of exogenous variables by $U = (U_{X_j} : j)$. These variables in U are never observed and are not affected by the endogenous variables in the model, but instead they affect the endogenous variables. They may also be referred to as background or error variables. One assumes that X_j is some function of $Pa(X_j)$ and an exogenous U_{X_j}:

$$X_j = f_{X_j}(Pa(X_j), U_{X_j}), \ j = 1 \ldots, J.$$

The collection of functions f_{X_j} indexed by all the endogenous variables is represented by $f = (f_{X_j} : j)$. Together with the joint distribution of U, these functions f_{X_j}, specify the data-generating distribution of (U, X) as they describe a deterministic system of structural equations (one for each endogenous variable X_j) that deterministically maps a realization of U into a realization of X. In an SCM one also refers to some of the endogenous variables as intervention variables. The SCM assumes that intervening on one of the intervention variables by setting their value, thereby making the function for that variable obsolete, does not change the form of the other functions. The functions f_{X_j} are often unspecified, but in some cases it might be reasonable to assume that these functions have to fall in a certain more restrictive class of functions. Similarly, there might be some knowledge about the joint distribution of U. The set of possible data-generating distributions of (U, X) can be obtained by varying the structural equations f over all allowed forms, and the distribution of the errors U over all possible error distributions defines the SCM for the full-data (U, X), i.e., the SCM is a statistical model for the random variable (U, X). An example of a fully parametric SCM would be obtained by assuming that all the functions f_{X_j} are known up to a finite number of parameters and that the error distribution is a multivariate normal distribution with mean zero and unknown covariance matrix. Such

parametric structural equation models are not recommended, for the same reasons as outlined in Chap. 1.

The corresponding SCM for the observed data O also includes specifying the relation between the random variable (U, X) and the observed data O, so that the SCM for the full data implies a parameterization of the probability distribution of O in terms of f and the distribution P_U of U. This SCM for the observed data also implies a statistical model for the probability distribution of O.

Let's translate these concepts into our mortality study example. We have the functions $f = (f_W, f_A, f_Y)$ and the exogenous variables $U = (U_W, U_A, U_Y)$. The values of W, A, and Y are deterministically assigned by U corresponding to the functions f. We specify our structural equation models, based on investigator knowledge, as

$$\begin{aligned} W &= f_W(U_W), \\ A &= f_A(W, U_A), \\ Y &= f_Y(W, A, U_Y), \end{aligned} \tag{2.1}$$

where no assumptions are made about the true shape of f_W, f_A, and f_Y. These functions f are nonparametric as we have not put a priori restrictions on their functional form. We may assume that U_A is independent of U_Y, given W, which corresponds with believing that there are no unmeasured factors that predict both A and the outcome Y: this is often called the no unmeasured confounders assumption. This SCM represents a semiparametric statistical model for the probability distribution of the errors U and endogenous variables $X = (W, A, Y)$. We assume that the observed data structure $O = (W, A, Y)$ is actually a realization of the endogenous variables (W, A, Y) generated by this system of structural equations. This now defines the SCM for the observed data O. It is easily seen that any probability distribution of O can be obtained by selecting a particular data-generating distribution of (U, X) in this SCM. Thus, the statistical model for P_0 implied by this SCM is a nonparametric model. As a consequence, one cannot determine from observing O if the assumptions in the SCM contradict the data. One states that the SCM represents a set of nontestable causal assumptions we have made about how the data were generated in nature.

Specifically, with the SCM represented in (2.1), we have assumed that the underlying data were generated by the following actions:

1. Drawing unobservable U from some probability distribution P_U ensuring that U_A is independent of U_Y, given W,
2. Generating W as a deterministic function of U_W,
3. Generating A as a deterministic function of W and U_A,
4. Generating Y as a deterministic function of W, A, and U_Y.

What if, instead, our SCM had been specified as follows:

$$\begin{aligned} W &= f_W(U_W), \\ A &= f_A(U_A), \\ Y &= f_Y(W, A, U_Y). \end{aligned} \tag{2.2}$$

What different assumption are we making here? If you compare (2.1) and (2.2), you see that the only difference between the two is the structural equation for f_A. In (2.2), A is evaluated as a deterministic function of U_A only. The baseline variables W play no role in the generation of variable A. We say that (2.2) is a more restrictive SCM than (2.1) because of this additional assumption about data generation. When might a researcher make such an assumption? In Chap. 1, we discussed RCTs. RCTs are studies where the subjects are randomized to treatment in the study. If our study of LTPA had been an RCT, it would make sense to assume the SCM specified in (2.2) given our knowledge of the study design. However, since it would be unethical to randomize subjects to levels of exercise, given the known health benefits, our study of LTPA on mortality is observational and we assume the less restrictive (2.1).

Causal assumptions made by the SCM for the full data:

- For each endogenous X_j, $X_j = f_j(Pa(X_j), U_{X_j})$ only depends on the other endogenous variables through its parents $Pa(X_j)$.
- The exogenous variables have a particular joint distribution P_U.

The SCM for the observed data includes the following additional assumption:

- The probability distribution of observed data structure O is implied by the probability distribution of (U, X).

After having specified the parent sets $Pa(X_j)$ for each endogenous variable X_j, one might make an assumption about the joint distribution of U, denoted P_U, representing knowledge about the underlying random variable (U, X) as accurately as possible. This kind of assumption would typically not put any restrictions on the probability distribution of O. The underlying data (U, X) are comprised of the exogenous variables U and the endogenous variables X, which is why we use the notation (U, X). In a typical SCM, the endogenous variables are the variables for which we have some understanding, mostly or fully observed, often collected according to a time ordering, and are very meaningful to the investigator. On the other hand, typically much of the distribution of U is poorly understood. In particular, one would often define U_{X_j} as some surrogate of potential unmeasured confounders, collapsing different poorly understood phenomena in the real world in one variable. The latter is reflected by the fact that we do not even measure these confounders, or know how to measure them. However, in some applications something about the joint distribution of U might be understood, and some components of U might be measured. For example, it might be known that treatment was randomized as in an RCT, implying that the error U_A for that treatment variable is independent of all other errors. On the other hand, in an observational study, one might feel uncomfortable making the assumption that U_A is independent of U_Y, given W, since one might know that some of the true confounders were not measured and are thereby captured by U_A.

Relationship of X and O. Our observed random variable O is related to X, and has a probability distribution that is implied by the distribution of (U, X). Specification of this relation is an important assumption of the SCM for the observed data O. A typical example is that $O = \Phi(X)$ for some Φ, i.e., O is a function of X. This includes the special case that $O \subset X$, i.e., with O being a simple subset of X. Because of this relationship $O = \Phi(X)$, the marginal probability distribution of X,

$$P_X(x) = \sum_u P_f(X = x \mid U = u) P_U(U = u),$$

also identifies the probability distribution of O through the functions $f = (f_{X_j} : j)$ and the distribution of the exogenous errors U. [Note that the conditional probability distribution $P_f(X = x \mid U = u)$ of X, given a realization $U = u$, is indeed completely determined by the functions f, which explains our notation P_f.] For example, if $X = O$, then:

$$P(o) = \sum_u P_f(X = o \mid U = u) P_U(U = u).$$

In order to make explicit that the probability distribution P of O is implied by the probability distribution of (U, X), we use the notation $P = P(P_{U,X})$. The true probability distribution $P_{U,X,0}$ of (U, X) implies the true probability distribution P_0 of O through this relation: $P_0 = P(P_{U,X,0})$. Since the assumed SCM often does not put any restrictions on the functions f_{X_j}, and the selection of the parent sets $Pa(X_j)$ might be purely based on time ordering (thereby not implying conditional independencies among the X_js), for many types of restrictions one would put on P_U, the resulting SCM for (U, X) would still not provide any restriction on the distribution of O. In that case, these causal assumptions provide no restriction on the distribution of O itself and thus imply a nonparametric *statistical* model $\mathcal{M}$ for the distribution P_0 of O. This statistical model $\mathcal{M}$ implied by the SCM for the observed data is given by $\mathcal{M} = \{P(P_{U,X}) : P_{U,X}\}$, where $P_{U,X}$ varies over all possible probability distributions of (U, X) in the SCM.

Each possible probability distribution $P_{U,X}$ of (U, X) in the SCM for the full data, indexed by a choice of error distribution P_U and a set of deterministic functions $(f_{X_j} : j)$, implies a probability distribution $P(P_{U,X})$ of O. In this manner the SCM for the full data implies a parameterization of the true probability distribution of O in terms of a true probability distribution of (U, X), so that the statistical model $\mathcal{M}$ for the probability distribution P_0 of O can be represented as $\mathcal{M} = \{P(P_{U,X}) : P_{U,X}\}$, where $P_{U,X}$ varies over all allowed probability distributions of (U, X) in the SCM. If this statistical model $\mathcal{M}$ implied by the SCM is nonparametric, then it follows that none of the causal assumptions encoded by the SCM are testable from the observed data.

2.2 Causal Graphs

SCMs provide a system for assigning values to a set of variables from random input. They are also an effective and straightforward means for explicitly specifying causal assumptions and the identifiability of the causal parameter of interest based on the observed data. We can draw a causal graph from our SCM, which is a visual way to describe some of the assumptions made by the model and the restrictions placed on the joint distribution of the data (U, X). However, in this text we do not place heavy emphasis on causal graphs as their utility is limited in many situations (e.g., complicated longitudinal data structures), and simpler visual displays of time ordering may provide more insight. Causal graphs also cannot encode every assumption we make in our SCM, and, in particular, the identifiability assumptions derived from causal graphs alone are not specific for the causal parameter of interest. Identifiability assumptions derived from a causal graph will thus typically be stronger than required. In addition, the link between the observed data and the full-data model represented by the causal graph is often different than simply stating that O corresponds with observing a subset of all the nodes in the causal graph. In this case, the causal graph itself cannot be used to assess the identifiability of a desired causal parameter from the observed data distribution.

2.2.1 Terminology

Figure 2.1 displays a possible causal graph for (2.1). The graph is drawn based on the relationships defined in f. The parents $Pa(X_j)$ of each X_j are connected to each X_j with an arrow directed toward X_j. Each X_j also has a directed arrow connecting its U_{X_j}. For example, the parents of Y, those variables in X on the right-hand side of the equation f_Y, are A and W. In Fig. 2.1, A and W are connected to Y, the child, with directed arrows, as is the exogenous U_Y. The baseline covariates W are represented with one variable. All the variables X and U in the graph are called nodes, and the lines that connect nodes are edges. All ancestors of a node occur before that node and all descendants occur after that node. This is a directed graph, meaning that each edge has only one arrow.

A path is any sequence of edges in a graph connecting two nodes. An example of a directed path in Fig. 2.1 is $W \rightarrow A \rightarrow Y$. This path connects each node with arrows that point in the direction of the path. In this figure there are several backdoor paths, which are paths that start with a node that has a directed arrow pointing into that node. The path can then be followed without respect to the direction of the arrows. For example, the path from Y to A through W is a backdoor path. Likewise, the path from Y to W through A is a backdoor path. These graphs are also acyclic; you cannot start at a node in a directed path and then return back to the same node through a closed loop. A collider is a node in a path where both arrows are directed toward the node. There are no colliders in Fig. 2.1. A blocked path is any path with at least one collider. A direct effect is illustrated by a directed arrow between two nodes, with

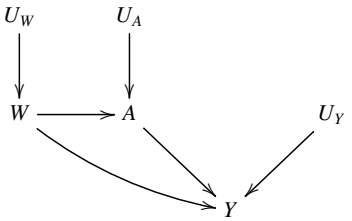

Fig. 2.1 A possible causal graph for (2.1).

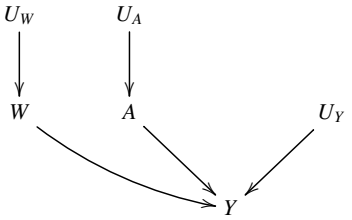

Fig. 2.2 A possible causal graph for (2.2)

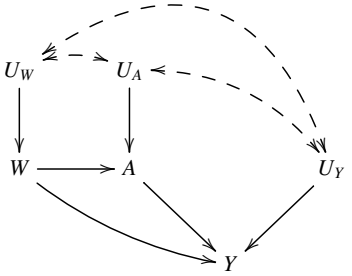

Fig. 2.3 A causal graph for (2.1) with no assumptions on the distribution of P_U

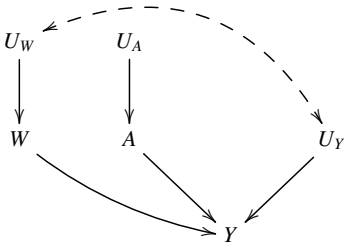

Fig. 2.4 A causal graph for (2.2) with no assumptions on the relationship between U_W and U_Y

no nodes mediating the path. Any unblocked path from A to Y other than the direct effect connecting A and Y represents an indirect effect of A on Y. One must block all unblocked backdoor paths from A to Y in order to isolate the causal effect of A on Y.

2.2.2 Assumptions

In Sect. 2.1, we discussed the typically nontestable causal assumptions made by an SCM. We make the first assumption by defining the parents $Pa(X_j)$ for each endogenous X_j. The second is any set of assumptions about the joint distribution P_U of the exogenous variables.

The assumptions made based on actual knowledge concerning the relationships between variables [i.e., defining the parents $Pa(X_j)$ for each endogenous X_j] are displayed in our causal graph through the presence and absence of directed arrows. The explicit absence of an arrow indicates a known lack of a direct effect. In many cases all arrows are included as it is not possible to exclude a direct effect based on a priori knowledge. In Fig. 2.1, the direction of the arrows is defined by the assignment of the parents to each node, including the time ordering assumed during the specification of (2.1). There is no explicit absence of any arrows; no direct effects are excluded. However, if we were to draw a graph for (2.2), it could look like Fig. 2.2. The direct effect between W and A is excluded because A is evaluated as a deterministic function of U_A only.

The assumptions on the distribution P_U are reflected in causal graphs through dashed double-headed arrows between the variables U. In Figs. 2.1 and 2.2, there are no arrows between the $U = (U_W, U_A, U_Y)$. Therefore, (2.1) and (2.2) included the assumption of joint independence of the endogenous variables U, which is graphically displayed by the lack of arrows. This is not an assumption one is usually able to make based on actual knowledge. More likely, we are able to make few or no assumptions about the distribution of P_U.

For (2.1), with no assumptions about the distribution of P_U, our causal graph would appear as in Fig. 2.3. For (2.2), our causal graph based on actual knowledge may look like Fig. 2.4. Since A is randomized, this implies that U_A is independent of U_Y and U_W, and we remove the arrows connecting U_A to U_Y and U_A to U_W. However, we have no knowledge to indicate the independence of U_Y and U_W, thus we cannot remove the arrows between these two variables.

The causal graph encodes some of the information and assumptions described by the SCM. It is an additional tool to visually describe assumptions encoded by the SCM. In more complex longitudinal data structures, it may be simpler to work with the SCM over the causal graph, as the intricacies of the causal relationships and abundance of arrows can limit the utility of the graphic.

2.3 Defining the Causal Target Parameter

Now that we have a way of modeling the data-generating mechanism with an SCM, we can focus on what we are trying to learn from the observed data. That is, we can define a causal target parameter of interest as a parameter of the distribution of the full-data (U, X) in the SCM. Formally, we denote the SCM for the full-data (U, X) by $\mathcal{M}^F$, a collection of possible $P_{U,X}$ as described by the SCM. In other words, $\mathcal{M}^F$, a model for the full data, is a collection of possible distributions for the underlying data (U, X). Ψ^F is a mapping applied to a $P_{U,X}$ giving $\Psi^F(P_{U,X})$ as the target parameter of $P_{U,X}$. This mapping needs to be defined for each $P_{U,X}$ that is a possible distribution of (U, X), given our assumptions coded by the posed SCM. In this way, we state $\Psi^F : \mathcal{M}^F \rightarrow \mathbb{R}^d$, where $\mathbb{R}^d$ indicates that our parameter is a vector of d real numbers. The SCM $\mathcal{M}^F$ consists of the distributions indexed by the deterministic function $f = (f_{X_j} : j)$ and distribution P_U of U, where f and this joint distribution P_U are identifiable from the distribution of the full-data (U, X). Thus the target parameter can also be represented as a function of f and the joint distribution of U.

Recall our mortality example with data structure $O = (W, A, Y)$ and SCM given in (2.1) with no assumptions about the distribution P_U. We can define $Y_a = f_Y(W, a, U_Y)$ as a random variable corresponding with intervention $A = a$ in the SCM. The marginal probability distribution of Y_a is thus given by

$$P_{U,X}(Y_a = y) = P_{U,X}(f_Y(W, a, U_Y) = y).$$

The causal effect of interest for a binary A (suppose it is the causal risk difference) could then be defined as a parameter of the distribution of (U, X) given by

$$\Psi^F(P_{U,X}) = E_{U,X}Y_1 - E_{U,X}Y_0.$$

In other words, $\Psi^F(P_{U,X})$ is the difference of marginal means of counterfactuals Y_1 and Y_0. We discuss this in more detail in the next subsection.

2.3.1 Interventions

We will define our causal target parameter as a parameter of the distribution of the data (U, X) under an intervention on one or more of the structural equations in f. The intervention defines a random variable that is a function of (U, X), so that the target parameter is $\Psi^F(P_{U,X})$. In Chap. 1, we discussed the "ideal experiment" which we cannot conduct in practice, where we observe each subject's outcome at all levels of A under identical conditions. Intervening on the system defined by our SCM describes the data that would be generated from the system at the different levels of our intervention variable (or variables). For example, in our study of LTPA on mortality, we can intervene on the exposure LTPA in order to observe the results

of this intervention on the system. By assumption, intervening and changing the functions f_{X_j} of the intervention variables does not change the other functions in f. With the SCM given in (2.1) we can intervene on f_A and set $a = 1$:

$$\begin{aligned} W &= f_W(U_W), \\ a &= 1, \\ Y_1 &= f_Y(W, 1, U_Y). \end{aligned}$$

We can also intervene and set $a = 0$:

$$\begin{aligned} W &= f_W(U_W), \\ a &= 0, \\ Y_0 &= f_Y(W, 0, U_Y). \end{aligned}$$

The intervention defines a random variable that is a function of (U, X), namely, $Y_a = Y_a(U)$ for $a = 1$ and $a = 0$. The notation $Y_a(U)$ makes explicit that Y_a is random only through U. The probability distribution of the (X, U) under an intervention is called the postintervention distribution. Our target parameter is a parameter of the postintervention distribution of Y_0 and Y_1, i.e., it is a function of these two postintervention distributions, namely, some difference. Thus, the SCM for the full data allows us to define the random variable $Y_a = f_Y(W, a, U_Y)$ for each a, where Y_a represents the outcome that would have been observed under this system for a particular subject under exposure a. Thus, with the SCM we can carry out the "ideal experiment" and define parameters of the distribution of the data generated in this perfect experiment, even though our observed data are only the random variables $O_1, \ldots, O_n$.

Formally, and more generally, the definition of the target parameter involves first specifying a subset of the endogenous nodes X_j playing the role of intervention nodes. Let A_s denote the intervention nodes, $s = 0, \ldots, S$, so that $A = (A_s : s = 1, \ldots, S)$, which, in shorthand notation, we also denote by $A = (A_s : s)$. We will denote the other endogenous nodes in X by $L = (L_r : r)$. Thus, $X = ((A_s : s), (L_r : r))$. Static interventions on the A-nodes correspond with setting A to a fixed value a, while dynamic interventions deterministically set A_s according to a fixed rule applied to the parents of A_s. Static interventions are a subset of the dynamic interventions. We will denote such a rule for assigning d to the intervention nodes, but it should be observed that d defines a rule for each A_s. Thus $d = (d_s : s = 1, \ldots, S)$ is a set of S rules. Such rules d are also called dynamic treatment regimens.

For a particular intervention d on the A nodes, and for a given realization u, the SCM generates deterministically a corresponding value for L, obtained by erasing the f_{A_s} functions, and carrying out the intervention d on A in the parent sets of the remaining equations. We denote the resulting realization by $L_d(u)$ and note that $L_d(u)$ is implied by f and u. The actual random variable $L_d(U)$ is called a postintervention random variable corresponding with the intervention that assigns the intervention nodes according to rule d. The probability distribution of $L_d(U)$ can be described as

$$P(L_d(U) = l) = \sum_u P_f(L_d(u) = l \mid U = u)P_U(u) = \sum_u I(L_d(u) = l)P_U(u).$$

In other words, it is the probability that U falls in the set of u-realizations under which the SCM system deterministically sets $L_d(u) = l$. Indicator $I(L_d(u) = l)$ is uniquely determined by the function specifications f_{X_j} for the X_j nodes that comprise L. This shows explicitly that the distribution of $L_d(U)$ is a parameter of f and the distribution of U, and thus a well-defined parameter on the full-data SCM $\mathcal{M}^F$ for the distribution of (U, X). We now define our target parameter $\Psi^F(P_{U,X})$ as some function of $(P_{L_d} : d)$ for a set of interventions d. Typically, we define our target parameter as a so-called causal contrast that involves a difference between two of such d-specific postintervention probability distributions. This target parameter is referred to as a causal parameter since it is a parameter of the postintervention distribution of L as a function of an intervention choice on $A = (A_s : s)$ across one or more interventions.

2.3.2 Counterfactuals

We would ideally like to see each individual's outcome at all possible levels of exposure A. The study is only capable of collecting Y under one exposure, the exposure the subject experiences. We discussed interventions on our SCM in Sect. 2.3.1 and we intervened on A to set $a = 1$ and $a = 0$ in order to generate the outcome for each subject under $A = a$ in our mortality study. Recall that Y_a represents the outcome that would have been observed under this system for a particular subject under exposure a. For our binary exposure LTPA , we have $(Y_a : a)$, with $a \in \mathcal{A}$, and where $\mathcal{A}$ is the set of possible values for our exposure LTPA. Here, this set is simply $\{0, 1\}$, but in other examples it could be continuous or otherwise more complex. Thus, in our example, for each realization u, which might correspond with an individual randomly drawn from some target population, by intervening on (2.1), we can generate so-called counterfactual outcomes $Y_1(u)$ and $Y_0(u)$. These counterfactual outcomes are implied by our SCM; they are consequences of it. That is, $Y_0(u) = f_Y(W, 0, u_Y)$, and $Y_1(u) = f_Y(W, 1, u_Y)$, where $W = f_W(u_W)$ is also implied by u. The random counterfactuals $Y_0 = Y_0(U)$ and $Y_1 = Y_1(U)$ are random through the probability distribution of U. Now we have the expected outcome had everyone in the target population met or exceeded recommended levels of LTPA, and the expected outcome had everyone had levels of LTPA below health recommendations. For example, the expected outcome of Y_1 is the mean of $Y_1(u)$ with respect to the probability distribution of U. Our target parameter is a function of the probability distributions of these counterfactuals: $E_0Y_1 - E_0Y_0$.

2.3.3 Establishing Identifiability

Are the assumptions we have already made enough to express the causal parameter of interest as a parameter of the probability distribution P_0 of the observed data? We want to be able to write $\Psi^F(P_{U,X,0})$ as $\Psi(P_0)$ for some parameter mapping Ψ, where we remind the reader that the SCM also specifies how the distribution P_0 of the observed data structure O is implied by the true distribution $P_{U,X,0}$ of (U, X). Since the true probability distribution of (U, X) can be any element in the SCM $\mathcal{M}^F$, and each such choice $P_{U,X}$ implies a probability distribution $P(P_{U,X})$ of O, this requires that we show that $\Psi^F(P_{U,X}) = \Psi(P(P_{U,X}))$ for all $P_{U,X} \in \mathcal{M}^F$.

This step involves establishing possible additional assumptions on the distribution of U, or sometimes also on the deterministic functions f, so that we can identify the target parameter from the observed data distribution. Thus, for each probability distribution of the underlying data (U, X) satisfying the SCM with these possible additional assumptions on P_U, we have $\Psi^F(P_{U,X}) = \Psi(P(P_{U,X}))$ for some Ψ. O is implied by the distribution of (U, X), such as $O = X$ or $O \subset X$, and $P = P(P_{X,U})$, where $P(P_{U,X})$ is a distribution of O implied by $P_{U,X}$.

Let us denote the resulting full-data SCM by $\mathcal{M}^{F*} \subset \mathcal{M}^F$ to make clear that possible additional assumptions were made that were driven purely by the identifiability problem, not necessarily reflecting reality. To be explicit, $\mathcal{M}^F$ is the full-data SCM under the assumptions based on real knowledge, and $\mathcal{M}^{F*}$ is the full-data SCM under possible additional causal assumptions required for the identifiability of our target parameter. We now have that for each $P_{U,X} \in \mathcal{M}^{F*}$, $\Psi^F(P_{U,X}) = \Psi(P)$, with $P = P(P_{U,X})$ the distribution of O implied by $P_{U,X}$ (whereas P_0 is the true distribution of O implied by the true distribution $P_{U,X,0}$).

Theorems exist that are helpful to establish such a desired identifiability result. For example, if $O = X$, and the distribution of U is such that, for each s, A_s is independent of L_d, given $Pa(A_s)$, then the well-known g-formula expresses the distribution of L_d in terms of the distribution of O:

$$P(L_d = l) = \prod_{r=1}^{R} P(L_r = l_r \mid Pa_d(L_r)) = Pa_d(l_r)),$$

where $Pa_d(L_r)$ are the parents of L_r with the intervention nodes among these parent nodes deterministically set by intervention d.

This so-called sequential randomization assumption can be established for a particular independence structure of U by verifying the backdoor path criterion on the corresponding causal graph implied by the SCM and this independence structure on U. The backdoor path criterion states that for each A_s, each backdoor path from A_s to an L_r node that is realized after A_s is blocked by one of the other L_r nodes.

In this manner, one might be able to generate a number of independence structures on the distribution of U that provide the desired identifiability result. That is, the resulting model for U that provides the desired identifiability might be represented as a union of models for U that assume a specific independence structure.

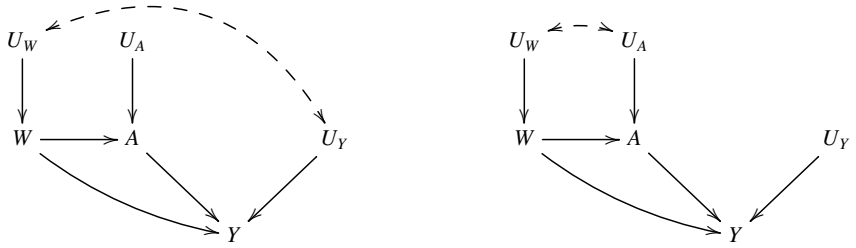

Fig. 2.5 Causal graphs for (2.1) with various assumptions about the distribution of P_U

If there is only one intervention node, i.e., $S = 1$, so that $O = (W, A, Y)$, the sequential randomization assumption reduces to the randomization assumption. The randomization assumption states that treatment node A is independent of counterfactual Y_a, conditional on W: $Y_a \perp A \mid Pa(A) = W$. You may be familiar with the (sequential) randomization assumption by another name, the no unmeasured confounders assumption. For our purposes, confounders are those variables in X one needs to observe in O in order to establish the identifiability of the target parameter of interest. We note that different such subsets of X may provide a desired identifiability result.

If we return to our mortality example and the structural equation models found in (2.1), the union of several independence structures allows for the identifiability of our causal target parameter $E_0 Y_1 - E_0 Y_0$ by meeting the backdoor path criterion. The independence structure in Fig. 2.3 does not meet the backdoor path criterion, but the two in Fig. 2.5 do. Thus in these two graphs the randomization assumption holds: A and Y_a are conditionally independent given W, which is implied by U_A being independent of U_Y, given W. It should be noted that Fig. 2.1 is a special case of the first graph in Fig. 2.5, so the union model for the distribution of U only represents two conditional independence models.

2.3.4 Commit to a Statistical Model and Target Parameter

The identifiability result provides us with a purely statistical target parameter $\Psi(P_0)$ on the distribution P_0 of O. The full-data model $\mathcal{M}^{F*}$ implies a statistical observed data model $\mathcal{M} = \{P(P_{X,U}) : P_{X,U} \in \mathcal{M}^{F*}\}$ for the distribution $P_0 = P(P_{U,X,0})$ of O. This now defines a target parameter $\Psi : \mathcal{M} \to \mathbb{R}^d$. The statistical observed data model for the distribution of O might be the same for $\mathcal{M}^F$ and $\mathcal{M}^{F*}$. If not, then one might consider extending the Ψ to the larger statistical observed data model implied by $\mathcal{M}^F$, such as possibly a fully nonparametric model allowing for all probability distributions. In this way, if the more restricted SCM holds, our target parameter would still estimate the target parameter, but one now also allows the data to contradict the more restricted SCM based on additional doubtful assumptions.

We can return to our example of the effect of LTPA on mortality and define our parameter, the causal risk difference, in terms of the corresponding statistical parameter $\Psi(P_0)$:

$$\Psi^F(P_{U,X,0}) = E_0Y_1 - E_0Y_0 = E_0[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)] \equiv \Psi(P_0),$$

where the outer expectation in the definition of $\Psi(P_0)$ is the mean across the strata for W. This identifiability result for the additive causal effect as a parameter of the distribution P_0 of O required making the randomization assumption stating that A is independent of the counterfactuals (Y_0, Y_1) within strata of W. This assumption might have been included in the original SCM $\mathcal{M}^F$, but, if one knows there are unmeasured confounders, then the model $\mathcal{M}^{F*}$ would be more restrictive by enforcing this "known to be wrong" randomization assumption.

Another required assumption is that $P_0(A = 1, W = w) > 0$ and $P_0(A = 0, W = w) > 0$ are positive for each possible realization w of W. Without this assumption, the conditional expectations of Y in $\Psi(P_0)$ are not well defined. This positivity assumption is often called the experimental treatment assignment (ETA) assumption. Here we are assuming that the conditional treatment assignment probabilities are positive for each possible w: $P_0(A = 1 \mid W = w) > 0$ and $P_0(A = 0 \mid W = w) > 0$ for each possible w. However, the positivity assumption is a more general name for the condition that is necessary for the target parameter $\Psi(P_0)$ to be well defined, and it often requires the censoring or treatment mechanism to have certain support.

So, to be very explicit about how this parameter corresponds with mapping P_0 into a number, as presented in Chap. 1:

$$\Psi(P_0) = \sum_w \Big[\sum_y yP_0(Y = y \mid A = 1, W = w) - \sum_y yP_0(Y = y \mid A = 0, W = w) \Big] P_0(W = w),$$

where

$$P_0(Y = y \mid A = a, W = w) = \frac{P_0(W = w, A = a, Y = y)}{\sum_y P_0(W = w, A = a, Y = y)}$$

is the conditional probability distribution of $Y = y$, given $A = a, W = w$, and

$$P_0(W = w) = \sum_{y,a} P_0(Y = y, A = a, W = w)$$

is the marginal probability distribution of $W = w$. This statistical parameter Ψ is defined on all probability distributions of (W, A, Y). The statistical model $\mathcal{M}$ is nonparametric and $\Psi : \mathcal{M} \to \mathbb{R}$.

We note again that we use the term statistical model for the collection of possible probability distributions, while we use the word model for the statistical model augmented with the nontestable causal assumptions coded by the underlying SCM and its relation to the observed data distribution of O. In our LTPA example, the model is the nonparametric statistical model augmented with the nontestable SCM. If this model includes the randomization assumption, and the experimental treatment assignment assumption, then this model allows the identifiability of the additive causal effect $E_0Y_1 - E_0Y_0$ through the statistical target parameter $\Psi(P_0) = E_0(E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W))$.

2.3.5 Interpretation of Target Parameter

The observed data parameter $\Psi(P_0)$ can be interpreted in two possibly distinct ways:

1. $\Psi(P_0)$ with $P_0 \in \mathcal{M}$ augmented with the truly reliable additional nonstatistical assumptions that are known to hold (e.g., $\mathcal{M}^F$). This may involve bounding the deviation of $\Psi(P_0)$ from the desired target causal effect $\Psi^F(P_{U,X,0})$ under a realistic causal model $\mathcal{M}^F$ that is not sufficient for the identifiability of this causal effect.
2. The truly causal parameter $\Psi^F(P_{U,X}) = \Psi(P_0)$ under the more restricted SCM $\mathcal{M}^{F*}$, thereby now including all causal assumptions that are needed to make the desired causal effect identifiable from the probability distribution P_0 of O.

The purely statistical (noncausal) parameter given by interpretation 1 is often of interest, such as $E_W[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, which can be interpreted as the average of the difference in means across the strata for W. With this parameter we can assume nothing, beyond the experimental treatment assignment assumption, except perhaps time ordering $W \rightarrow A \rightarrow Y$, to have a meaningful interpretation of the difference in means. Since we do not assume an underlying system, the SCM for (U, X) and thereby Y_a, or the randomization assumption, the parameter is a statistical parameter only. This type of parameter is sometimes referred to as a variable importance measure.

For example, if A = age, the investigator may not be willing to assume an SCM defining interventions on age (a variable one cannot intervene on and set in practice). Thus, if one does not assume $\mathcal{M}^F$, the statistical parameter $\Psi(P_0)$ under interpretation 1 can still be very much of interest. In some cases, however, these two interpretations coincide. What is known about the generation of data and distribution

P_U may imply the assumptions necessary to interpret $\Psi(P_0)$ as the causal parameter $\Psi^F(P_{U,X})$: for example, in an RCT, by design, assuming full compliance and no missingness or censoring, the causal assumptions required will hold.

2.4 Revisiting the Mortality Example

For the sake of presentation, we intentionally assumed that the exposure LTPA was binary and worked with an SCM that generated a binary exposure A. In the actual mortality study A is continuous valued. Consider the more realistic SCM $W = f_W(U_W), A = f_A(W, U_A), Y = f_Y(W, A, U_Y)$, where A is now continuous valued. Let $Y_a(u)$ be the counterfactual obtained by setting $A = a$ and $U = u$, so that Y_a is the random variable representing survival at 5 years under LTPA at level a. Suppose one wishes to consider a cut-off value δ for LTPA level so that one can recommend that the population at least exercise at this level δ. A causal quantity of interest is now

$$\psi_0^F = \sum_a w_1(a) E_0 Y_a - \sum_a w_0(a) E_0 Y_a,$$

where $w_1(a)$ is a probability distribution on excercise levels larger than δ, and $w_0(a)$ is a probability distribution on exercise levels smaller than or equal to δ. This corresponds to $E_0 Y_1 - E_0 Y_0$, where Y_1 is defined by the random intervention on the SCM in which one randomly draws A from w_1, and similarly Y_0 is defined by randomly drawing A from w_0. This causal effect $E_0 Y_1 - E_0 Y_0$ can be identified from the probability distribution P_0 of $O = (W, A, Y)$ as follows:

$$\psi_0^F = \sum_a (w_1 - w_0)(a) E_0 E_0(Y \mid A = a, W) \equiv \psi_0.$$

2.5 Road Map for Targeted Learning

In Chap. 1, we introduced the road map for targeted learning. In this chapter we have discussed defining the research question, which involved describing the data and committing to a statistical model and target parameter. The estimation problem we wish to solve is now fully defined. The next stage of the road map addresses estimation of the target parameter, which will be covered in the next three chapters.

The statistical estimation problem. We observe n i.i.d. copies $O_1, .., O_n$ from a probability distribution P_0 known to be in a statistical model $\mathcal{M}$, and we wish to infer statistically about the target parameter $\Psi(P_0)$. Often, this target parameter only depends on P_0 through a relevant (infinite-dimensional) parameter $Q_0 = Q_0(P_0)$ of P_0, so that we can also write $\Psi(Q_0)$.

Targeted substitution estimator. We construct a substitution estimator $\Psi(Q_n^*)$ obtained by plugging in an estimator Q_n^* of Q_0. This involves super learning and

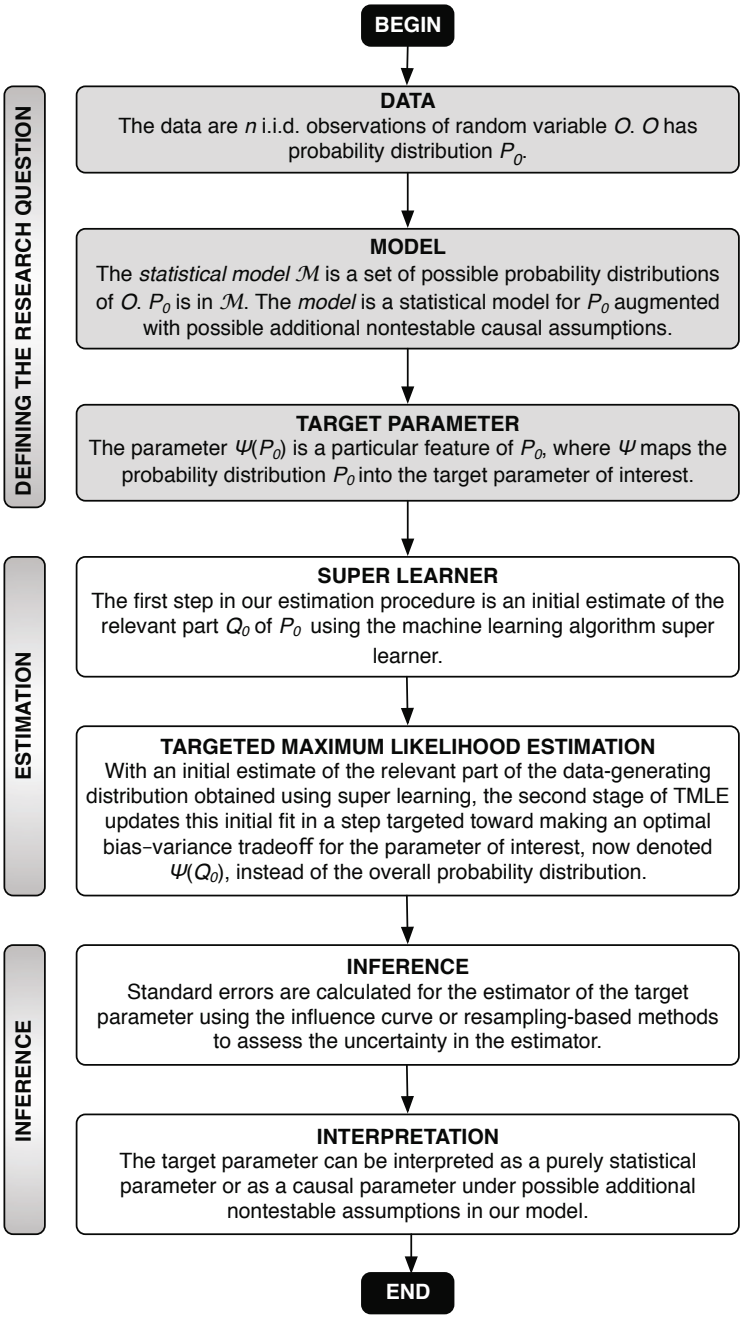

Fig. 2.6 Road map for targeted learning

TMLE, so that we obtain, under regularity conditions, an asymptotically linear, double robust, and efficient normally distributed estimator of $\psi_0 = \Psi(Q_0)$, and, in general, put in the maximal effort to minimize the mean squared error with respect to the true value ψ_0. In addition, we provide statistical inference about ψ_0 based on the estimation of the normal limit distribution of $\sqrt{n}(\Psi(Q_n^*) - \psi_0)$.

2.6 Conceptual Framework

This section provides a rigorous conceptual framework for the topics covered in this chapter. If you find it too abstract on your initial reading, we advise you to come back as you become more familiar with the material. It is meant for more advanced readers.

Data are meaningless without knowledge about the experiment that generated the data. That is, data are realizations of a random variable with a certain probability distribution on a set of possible outcomes, and statistical learning is concerned with learning something about the probability distribution of the data. Typically, we are willing to view our data as a realization of n independent identical replications of the experiment, and we accept this as our first modeling assumption. If we denote the random variable representing the data generated by the experiment by O, having a probability distribution P_0, then the data set corresponds with drawing a realization of n i.i.d. copies $O_1, \ldots, O_n$ with some common probability distribution P_0.

A statistical estimation problem corresponds with defining a statistical model $\mathcal{M}$ for P_0, where the statistical model $\mathcal{M}$ is a collection of possible probability distributions of O. The estimation problem also requires a mapping Ψ on this statistical model $\mathcal{M}$, where Ψ maps any $P \in \mathcal{M}$ into a vector of numbers $\Psi(P)$. We write $\Psi : \mathcal{M} \to \mathbb{R}^d$ for a d-dimensional parameter. We introduce ψ_0, and the interpretation of ψ_0 as $\Psi(P_0)$, i.e., a well-defined feature of P_0, is called the pure statistical interpretation of the parameter value ψ_0. The statistical estimation problem is now to map the data set $O_1, \ldots, O_n$ into an estimator of $\Psi(P_0)$ that incorporates the knowledge that $P_0 \in \mathcal{M}$, accompanied by an assessment of the uncertainty in the estimator of ψ_0.

When thinking purely about the construction of an estimator, the only concern is to construct an estimator of ψ_0 that has small mean squared error (MSE), or some other measure of dissimilarity between the estimator and the true ψ_0. This does not require any additional knowledge (or nontestable causal assumptions). As a consequence, for the construction of a targeted maximum likelihood estimator, which we introduce in Chaps. 4 and 5, the only input is the statistical model $\mathcal{M}$ and the mapping Ψ representing the target parameter.

Making assumptions about P_0 that do not change the statistical model, so-called nontestable assumptions, will not change the statistical estimation problem. However, such assumptions allows one to interpret a particular parameter $\Psi(P_0)$ in a new way. If such nontestable assumptions are known to be true, it enriches the interpretation of the number ψ_0. If they are wrong, then it results in misinterpretation of ψ_0.

This is called causal modeling when it involves nontestable assumptions that allow $\Psi(P_0)$ to be interpreted as a causal effect, and, in general, it is modeling with nontestable assumptions with the goal of providing an enriched interpretation of this parameter $\Psi(P_0)$.

It works as follows. One proposes a parameterization $\theta \rightarrow P_\theta$ for θ varying over a set Θ so that the statistical model $\mathcal{M}$ can be represented as $\mathcal{M} = \{P_\theta : \theta \in \Theta\}$, where θ represents $P_{U,X}$ in our SCM framework, but it can represent any underlying structure (not necessarily causal). That is, we provide a parameterization for the statistical model $\mathcal{M}$. In addition, since $P_0 \in \mathcal{M}$, there exists a θ_0 such that $P_0 = P_{\theta_0}$. Assume that this θ_0 is actually uniquely identified by P_0. θ_0 has its own interpretation, such as the probability distribution of counterfactual random variables in the SCM. Suddenly, the P_0 allows us to infer $\theta_0 = \Theta(P_0)$ for a mapping Θ. As a consequence, with this "magic trick" of parameterizing P_0 we succeeded in providing a new interpretation of P_0 and, in particular, of any parameter $\Psi(P_0) = \Psi(P_{\theta_0})$ as a function of θ_0.

As one can imagine, there are millions of possible magic tricks one can carry out, each one creating a new interpretation of P_0 by having it mapped into an interpretation of a θ_0 implied by a particular parameterization. The data cannot tell you if one magic trick will provide a more accurate description of reality than another magic trick, since data can only provide information about P_0 itself. As a consequence, which magic trick is applied, or if any trick is applied at all, should be driven by true knowledge about the underlying mechanism that resulted in the generation of O. In that case, the selection of the parameterization is not a magic trick but represents the incorporation of true knowledge allowing us to interpret the parameter ψ_0 for what it is. Note that this modeling could easily correspond with a nonparametric statistical model $\mathcal{M}$ for P_0.

Two important mistakes can occur in statistical practice, before the selection of an estimator, given that one has specified a statistical model $\mathcal{M}$ and parameter $\Psi : \mathcal{M} \rightarrow \mathbb{R}^d$. The first mistake is that one specifies the statistical model $\mathcal{M}$ incorrectly so that $P_0 \notin \mathcal{M}$, resulting in misinterpretation of $\Psi(P_0)$, even as a purely statistical parameter, i.e., as a mapping Ψ applied to P_0. The second mistake is that one misspecifies additional nontestable assumptions as coded by the selected parameterization for $\mathcal{M}$ that were used to provide an enriched interpretation of $\Psi(P_0)$, again resulting in misinterpretation of $\Psi(P_0)$. These two mistakes can be collapsed into one, namely, misspecification of the model for P_0. By the model we now mean the statistical model for P_0 augmented with the additional nontestable structural assumptions, even though these do not change the statistical model.

So a model now includes the additional parameterization, such that two identical statistical models that are based on different parameterizations are classified as different models. Thus, a model is defined by a mapping $P_\cdot : \Theta \rightarrow \mathcal{M}$, $\theta \rightarrow P_\theta$, and the statistical model implied by this model is given by the range $\mathcal{M} = \{P_\theta : \theta\}$ of this mapping. Regarding statistical vocabulary, we will use the word model for the parameterization mapping $P_\cdot : \Theta \rightarrow \mathcal{M}$, and statistical model for the set of possible probability distributions, i.e., the range of this mapping. Note, that if the parame-

terization is simply the identity mapping defined on $\mathcal{M}$, then the model equals the statistical model.

Even though it is healthy to be cynical about modeling and extremely aware of its dangers and its potential to lie with data, it is of fundamental importance to statistical learning that we can incorporate structural knowledge about the data-generating process and utilize that in our interpretation. In addition, even if these structural assumptions implied by the model/parameterization are uncertain, it is worthwhile to know that, *if* these were true, then our parameter would allow its corresponding interpretation. One could then report both the statistical interpretation, or the reliable statistical model interpretation, as well as the *if also, then* interpretation to our target ψ_0.

In addition, this structural modeling allows one to create truly interesting parameters in an underlying world and one can then establish under what assumptions one can identify these truly interesting parameters from the observed data. This itself teaches us how to generate new data so that these parameters will be identifiable. The identifiability results for these truly interesting parameters provide us with statistical parameters $\Psi(P_0)$ that might be interesting as statistical parameters anyway, without these additional structural assumptions, and have the additional flavor of having a particularly powerful interpretation if these additional structural assumptions happen to be true. In particular, one may be able to interpret $\Psi(P_0)$ as the best possible approximation of the wished causal quantity of interest based on the available data. Overall, this provides us with more than enough motivation to include (causal) modeling as an important component in the road map of targeted learning from data.

2.7 Notes and Further Reading

As noted in the introduction, a thorough presentation of SCMs, causal graphs, and related identifiability theory can be found in Pearl (2009). We also direct the interested reader to Judea Pearl's Web site (http://bayes.cs.ucla.edu/jp_home.html) for easily organized references and presentations on these topics. The g-formula for identifying the distribution of counterfactuals from the observed data distribution, under the sequential randomization assumption, was originally published in Robins (1986). The simplified data example we introduce in this chapter, a mortality study examining the effect of LTPA, is based on data presented in Tager et al. (1998). We carry this example through the next three chapters, and in Chap. 4, we analyze this data using super learning and targeted maximum likelihood estimation.

In our road map we utilize causal models, such as SCMs and the Neyman–Rubin model, to generate statistical effect parameters $\psi_0 = \Psi(P_0)$ of interest. The interpretation of the estimand ψ_0, beyond its pure statistical interpretation, depends on the required causal assumptions necessary for identifiability of the desired causal quantity ψ_0^F (defined as target quantity in causal model for full data or counterfactuals) from the observed data distribution. Such an interpretation might be further

enriched if one could define an actual experiment that would reproduce this causal quantity. Either way, our road map poses these causal models as working models to derive these statistical target parameters that can be interpreted as causal effects under explicitly stated causal assumptions. The latter assumptions are fully exposed and for anybody to criticize.

We wish to stress that the learning of these estimands with their pure statistical interpretation already represents progress in science. In addition, the required causal assumptions that would allow a richer interpretation of the estimand teach us how to improve our design of the observational or RCT.

Somehow, we think that a statistical target parameter that has a desired causal interpretation under possibly unrealistic assumptions is a "best" approximation of the ideal causal quantity, given the limitations set by the available data. For example, $E_0(E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W))$ is an effect of treatment, controlling for the measured covariates, with a clear statistical interpretation, and, if people feel comfortable talking about $E_0Y_1 - E_0Y_0$, then we think that this statistical estimand represents a "best" effort to target this additive causal effect under the constraints set by the available data.

Instead of making a hard decision regarding the causal assumptions necessary for making the estimand equal to the causal quantity, one may wish to investigate the potential distance between the estimand and the causal quantity. In this manner, one still allows for a causal interpretation of the estimand (such as that the asymptotic bias of the estimand with respect to the desired causal quantity is bounded from above by a certain number), even if the causal assumptions required for making the estimand equal to the causal quantity are violated. Such an approach relies on the ability to bound this distance by incorporation of realistic causal knowledge. Such a sensitivity analysis will require input from subject matter people such as a determination of an upper bound of the effect of unmeasured confounders beyond the measured time-dependent confounders. Even a highly trained statistician will have an extremely hard time getting his/her head around such a question, making such sensitivity analyses potentially unreliable and extremely hard to communicate. Still, this is an important research area since it allows for a continuous range from pure statistical interpretation of the estimand to a pure causal effect interpretation.

Either way, we should not forget that using poor methods for estimation with the actual observed data, while investing enormous effort in such a sensitivity analysis. makes no sense. By the same token, estimation of the estimand is a separate problem from determining the distance between the estimand and the causal quantity of interest and is obviously as important as carefully defining and interpreting the estimand: the careful definition and interpretation of an estimand has little value if one decides to use a misspecified parametric model to fit it!

Chapter 3
Super Learning

Eric C. Polley, Sherri Rose, Mark J. van der Laan

This is the first chapter in our text focused on estimation within the road map for targeted learning. Now that we've defined the research question, including our data, the model, and the target parameter, we are ready to begin. For the estimation of a target parameter of the probability distribution of the data, such as target parameters that can be interpreted as causal effects, we implement TMLE. The first step in this estimation procedure is an initial estimate of the data-generating distribution P_0, or the relevant part Q_0 of P_0 that is needed to evaluate the target parameter. This is the step presented in Chap. 3, and TMLE will be presented in Chaps. 4 and 5.

We introduce these concepts using our mortality study example from Chap. 2 examining the effect of LTPA. Our outcome Y is binary, indicating death within 5 years of baseline, and A is also binary, indicating whether the subject meets recommended levels of physical activity. The data structure in this example is $O = (W, A, Y) \sim P_0$. Our target parameter is $\Psi(P_0) = E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, which represents the causal risk difference under causal assumptions. Since this target parameter only depends on P_0 through the conditional mean $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$, and the marginal distribution $Q_{W,0}$ of W, we can also write $\Psi(Q_0)$, where $Q_0 = (\bar{Q}_0, Q_{W,0})$. We estimate the expectation over W with the empirical mean over W_i, $i = 1, \ldots, n$. With this target parameter, $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$ is the only object we will still need to estimate. Therefore, the first step of the TMLE of the risk difference $\Psi(P_0)$ is to estimate this conditional mean function $\bar{Q}_0(A, W)$. Our substitution TMLE will be of the type

$$\psi_n = \Psi(Q_n) = \frac{1}{n} \sum_{i=1}^{n} \{\bar{Q}_n(1, W_i) - \bar{Q}_n(0, W_i)\},$$

where this estimate is obtained by plugging $Q_n = (\bar{Q}_n, Q_{W,n})$ into the parameter mapping Ψ.

We could estimate the entire conditional probability distribution of Y, instead of estimating the conditional mean of Y, but then (except when Y is binary) we are estimating portions of the density we do not need. Targeted estimation of only the

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_3,

relevant portion of the probability distribution of O in this first step of the TMLE procedure provides us with maximally efficient and unbiased estimators. This will be further discussed in Chaps. 4 and 5.

3.1 Background

Let's start our discussion with studies where Y is binary, such as in our mortality study example. When Y is binary, there is no difference between the conditional mean or conditional probability distribution, so this distinction plays no role. Now, what do we know about our probability distribution P_0 of O? We know that the data are n i.i.d. observations (realizations) on n i.i.d. copies $O_1, \ldots, O_n$ of $O \sim P_0$. These realizations are denoted $o_1, \ldots, o_n$. In our mortality study example, we have no knowledge about P_0. Thus we have a nonparametric statistical model for P_0. In scenarios where we have some knowledge about data generation, we can include this knowledge in a semiparametric statistical model. Our parameter of interest is this chapter is $\bar{Q}_0(A, W) = P_0(Y = 1 \mid A, W)$.

How are we to estimate $P_0(Y = 1 \mid A, W)$ if we assume only a nonparametric (or, in general, a large semiparametric) statistical model? We do not know anything about the shape of $\bar{Q}_0(A, W)$ as a function of exposure and covariates. Standard practice would assume a parametric statistical model, making assumptions we know are wrong, and proceeding to estimate $P_0(Y = 1 \mid A, W)$ under the assumptions of the parametric statistical model, thereby forcing the shape of this function of (A, W) to follow an incorrect user-supplied structure. Since the parametric statistical model is wrong, the estimate of $P_0(Y = 1 \mid A, W)$ will be biased, and increasing the sample size will not make it any better. What we want is an automated algorithm to nonparametrically (or semiparametrically) estimate $P_0(Y = 1 \mid A, W)$, i.e., we want an estimator that is able to learn from the data using the true knowledge represented by the actual statistical model for P_0.

In the computer science literature, this is called machine learning. In statistics, these methods are often referred to as nonparametric or semiparametric estimators, or data-adaptive estimators. We will use the terms *data-adaptive* and *machine learning* interchangeably in this text. The essential point is that there are nonparametric methods that also aim to "smooth" the data and estimate this regression function flexibly, adapting it to the data given a priori guidelines, without overfitting the data.

For example, one could use local averaging of the outcome Y within covariate "neighborhoods." Here, neighborhoods are bins for covariate observations that are close in value, where these bins are defined by partitioning the covariate space. The number of bins will determine the smoothness of our fitted regression function. Such a regression estimator is also called a histogram regression estimator. How do you choose the size of these neighborhoods or bins? This becomes a bias–variance trade-off question. If we have many small neighborhoods, the estimate will not be smooth and will have high variance since some neighborhoods will be empty or contain only a small number of observations. The result is a sample mean of the outcome

over the observations in the neighborhood that is imprecise. On the other hand, if we have very few large neighborhoods, the estimate is much smoother, but it will be biased since the neighborhoods fail to capture the complexity of the data. Suppose we choose the number of neighborhoods in a smart way. With n large enough, this will result in a good estimator in our nonparametric statistical model. Formally, we say such a histogram regression estimator is asymptotically consistent in the sense that it approximates the true regression function as sample size increases.

However, if the true data-generating distribution is very smooth, a logistic regression in a misspecified parametric statistical model might beat the nonparametric estimator. This is frustrating! We want to create a smart nonparametric estimator that is consistent, but in some cases it may "lose" to a misspecified parametric model because it is more variable. There are other ways of approaching the truth that will be smoother than local averaging. One method, locally weighted regression and scatterplot smoothing (loess), is a weighted polynomial regression method that fits the data locally, iteratively within neighborhoods. Spline functions, another method, are similar to polynomial functions, as splines are piecewise polynomial functions. Smoothing splines use penalties to adjust for a lack of smoothness, and regression splines use linear combinations of basis functions. There are many other potential algorithms we could implement to estimate $P_0(Y = 1 \mid A, W)$. However, how are we to know priori which one to use? We cannot bet on a logistic regression in a misspecified parametric statistical model, but we have the problem that one particular algorithm is going to do better than the other candidate estimators for the particular data-generating distribution P_0, and we do not know which one is the best.

To be very explicit, an algorithm is an estimator of $\bar{Q}_0$ that maps a data set of n observations (W_i, A_i, Y_i), $i = 1, \ldots, n$, into a prediction function that can be used to map input (A, W) into a predicted value for Y. The algorithms may differ in the subset of the covariates used, the basis functions, the loss functions, the searching algorithm, and the range of tuning parameters, among others. We use *algorithm* in a general sense to mean any mapping from data into a predictor, so that the word *algorithm* is equivalent to the word *estimator*. As long as the algorithm takes the observed data and outputs a fitted prediction function, we consider it a prediction algorithm. For example, a collection of algorithms could include least squares regression estimators, algorithms indexed by set values of the fine-tuning parameters for a collection of values, algorithms using internal cross-validation to set fine-tuning parameters, algorithms coupled with screening procedures to reduce the dimension of the covariate vector, and so on.

Effect Estimation vs. Prediction

Both causal effect and prediction research questions are inherently *estimation* questions. In the first, we are interested in estimating the causal effect of A on Y adjusted for covariates W. For prediction, we are interested in generating a function to input the variables (A, W) and predict a value for Y. These are separate and distinct research questions. However, many (causal) effect esti-

mators, such as TMLE, involve prediction steps within the procedure. Thus, understanding prediction is a core concept even when one has an effect estimation research question. Effect parameters where no causal assumptions are made are often referred to as variable importance measures (VIMs).

3.2 Defining the Estimation Problem

Our data structure is $O = (W, A, Y) \sim P_0$, and we observe n i.i.d. observations on $O_1, \ldots, O_n$. An estimator maps these observations into a value for the parameter it targets. We can view estimators as mappings from the empirical distribution P_n of the data set, where P_n places probability $1/n$ on each observed O_i, $i = 1, \ldots, n$. In our mortality study example, we need an estimator of $\bar{Q}_0(A, W) = P_0(Y = 1 \mid A, W)$.

Before we can choose a "best" algorithm to estimate the function $\bar{Q}_0 : (A, W) \rightarrow \bar{Q}_0(A, W)$, we must have a way to define what "best" means. We do this in terms of a loss function, which assigns a measure of performance to a candidate function $\bar{Q}$ when applied to an observation O. That is, a loss function is a function L given by

$$L : (O, \bar{Q}) \rightarrow L(O, \bar{Q}) \in \mathbb{R}.$$

It is a function of the random variable O and parameter value $\bar{Q}$. Examples of loss functions include the L_1 absolute error loss function

$$L(O, \bar{Q}) = |Y - \bar{Q}(A, W)|,$$

the L_2 squared error (or quadratic) loss function

$$L(O, \bar{Q}) = (Y - \bar{Q}(A, W))^2,$$

and the negative log loss function for a binary Y

$$L(O, \bar{Q}) = -\log(\bar{Q}(A, W)^Y (1 - \bar{Q}(A, W))^{1-Y}).$$

A loss function defines a function $\bar{Q}_0$ that has the optimal expected performance with respect to that loss function among all candidate functions $\bar{Q}$. For example, the function $\bar{Q}_0$ that minimizes the expected absolute error, $\bar{Q} \rightarrow E_0|Y - \bar{Q}(A, W)|$, is the conditional median of Y, as a function of (A, W). On the other hand, the function that minimizes the expected squared error is the conditional mean of Y, while the function that minimizes the expected negative log loss function for a binary Y is the conditional probability distribution of Y, as a function of (A, W). For binary Y, both the L_2 loss and negative log loss target the same function $\bar{Q}_0(A, W) = P_0(Y = 1 \mid A, W)$.

We can now define our parameter of interest, $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$, as the minimizer of the expected squared error loss:

$$\bar{Q}_0 = \arg\min_{\bar{Q}} E_0 L(O, \bar{Q}),$$

where $L(O, \bar{Q}) = (Y - \bar{Q}(A, W))^2$. $E_0 L(O, \bar{Q})$, which we want to be small, evaluates the candidate $\bar{Q}$, and it is minimized at the optimal choice of $\bar{Q}_0$. We refer to expected loss as the risk. Thus we have a way to define the "best" algorithm. We want the estimator of the regression function $\bar{Q}_0$ whose realized value minimizes the expectation of the squared error loss function. If we have two estimates $\bar{Q}_n^a$ and $\bar{Q}_n^b$, then we prefer the estimator for which $\sum_o P_0(O = o)L(o, \bar{Q}_n)$ is smallest.

This makes sense intuitively. We want an estimator that is close to the true $\bar{Q}_0$ and the difference between the risk at a candidate $\bar{Q}$ and the risk at the true $\bar{Q}_0$ corresponds with an expected squared error between $\bar{Q}$ and $\bar{Q}_0$ across all values of (A, W):

$$E_0 L(O, \bar{Q}) - E_0 L(O, \bar{Q}_0) = E_0(\bar{Q} - \bar{Q}_0)^2(A, W).$$

Minimizing the expected loss will bring the chosen candidate closer to the true $\bar{Q}_0$ with respect to the dissimilarity measure implied by the loss function, namely, the difference of the risk at $\bar{Q}$ and the optimal risk at $\bar{Q}_0$. How do we find out which algorithm among a library of algorithms yields the smallest expected loss, or, equivalently, which one has the best performance with respect to the dissimilarity implied by the loss function?

3.3 Super (Machine) Learning

Let us return to our simplified mortality study example. The outcome Y is binary, indicating death within 5 years of baseline, and A is also binary, indicating whether the subject meets recommended levels of physical activity. The data structure is $O = (W, A, Y) \sim P_0$. For now let us consider only the covariates $W = \{W_1, W_2, W_3\}$. Age (W_1) is a continuous measure, gender (W_2) is binary, and chronic health history (W_3) is a binary measure indicating whether the subject has a chronic health condition at baseline. While we are ultimately interested in the effect of LTPA on death demonstrated in the next chapter, if we were strictly interested in a *prediction* research question, we could also include LTPA as a covariate in vector W.

Suppose there are three subject matter experts, and they each have a different proposal about the specification of a logistic regression in a parametric statistical model, incorporating their subject matter knowledge. The first believes a main terms statistical model is sufficient for the estimation of the prediction target parameter:

$$\bar{Q}_n^a(A, W) = P_n^a(Y = 1 \mid A, W) = \text{expit}\,(\alpha_{0,n} + \alpha_{1,n}A + \alpha_{2,n}W_1 + \alpha_{3,n}W_2 + \alpha_{4,n}W_3).$$

The second expert proposes including all covariates W and exposure A, as well as an interaction term between age and gender:

$$\bar{Q}_n^b(A, W) = \text{expit}\,(\alpha_{0,n} + \alpha_{1,n}A + \alpha_{2,n}W_1 + \alpha_{3,n}W_2 + \alpha_{4,n}W_3 + \alpha_{5,n}(W_1 \times W_2)).$$

The third expert wants to use a statistical model with main terms and age^2:

$$\bar{Q}_n^c(A, W) = \text{expit}\,(\alpha_{0,n} + \alpha_{1,n}A + \alpha_{2,n}W_1 + \alpha_{3,n}W_2 + \alpha_{4,n}W_3 + \alpha_{5,n}W_1^2).$$

The investigators would ideally like to run all three of these statistical models. Now that we've defined a criterion for the best estimator of $\bar{Q}_0$, how can we responsibly select the optimal estimator from a collection of algorithms, such as the collection of estimators $\bar{Q}_n^a$, $\bar{Q}_n^b$, and $\bar{Q}_n^c$?

3.3.1 Discrete Super Learner

We start by introducing discrete super learning, which will give us an estimate of the cross-validated risk for each algorithm. The entire data set (learning set) is divided into V groups of size $\sim n/V$. These groups are mutually exclusive and exhaustive sets. Our mortality data set has $n = 2066$ subjects. If we want to perform V-fold cross-validation in our discrete super learning procedure, using 10 folds, we will divide our data set into groups of size $\sim 2066/10$. (This gives us four groups with 206 subjects and six groups with 207 subjects.) We label each group from 1 to 10.

Let us focus first on understanding the procedure with just one of the regressions in the collection of algorithms. The observations in group 1 are set aside, and the first regression is fit on the remaining nine groups (called the training set). Then we take the observations in group 1 (called the validation set) and obtain predicted probabilities of death for these 206 or 207 observations using the regression fit on the training set. It is important to note that the observations in group 1 *were not included in the fitting process* and will only be used to evaluate the performance of the predictor that was obtained on the training sample. In this way, we have succeeded in obtaining predicted probabilities of death for approx. 10% of our data, where the prediction function used to obtain these predicted probabilities was fit based on the remaining 90% of data. At this stage, we calculate the estimated risk within the validation set using their predicted probabilities. This procedure is performed for all of the algorithms in the collection of algorithms, so that we have, at the end of the first fold, predicted probabilities for each of the three regressions. We also have an estimate of risk within the validation set (group 1) for each of the three regressions, calculated using their corresponding predicted probabilities.

We need to perform this procedure nine more times, so that each group has the opportunity to take on the role of the validation set and obtain predicted probabilities for each algorithm fit on the corresponding training set. Thus, the procedure continues until we have predicted probabilities of death for all 2066 subjects for each algorithm, and also estimated risk within each validation set for each algorithm. We then have 10 estimated risks for each of the three algorithms, and these risks are averaged across validation sets resulting in one estimated cross-validated risk for

each algorithm. The discrete super learner algorithm selects the algorithm with the smallest cross-validated risk. The algorithm with the smallest cross-validated risk is the "best" estimator according to our criterion: minimizing the estimated expected squared error loss function. See Fig. 3.1 for a diagram of this procedure.

We have now described a new algorithm that took as input the three algorithms. This new estimator is what we call the discrete super learner, and it is indexed by this collection of three algorithms.

By incorporating a rich collection of algorithms that vary in bias and degree of data-fitting, the cross-validation within the discrete super learner prevents overfitting and it also prevents selecting a fit that is too biased. There are many forms of cross-validation, and here we discussed V-fold cross-validation due to its low computational burden while still providing the desirable finite sample and asymptotic optimality properties, which will be discussed later. The collection of algorithms can be large and includes other algorithms besides parametric statistical models, for example, the collection of algorithms may include random forest algorithms and support vector machines.

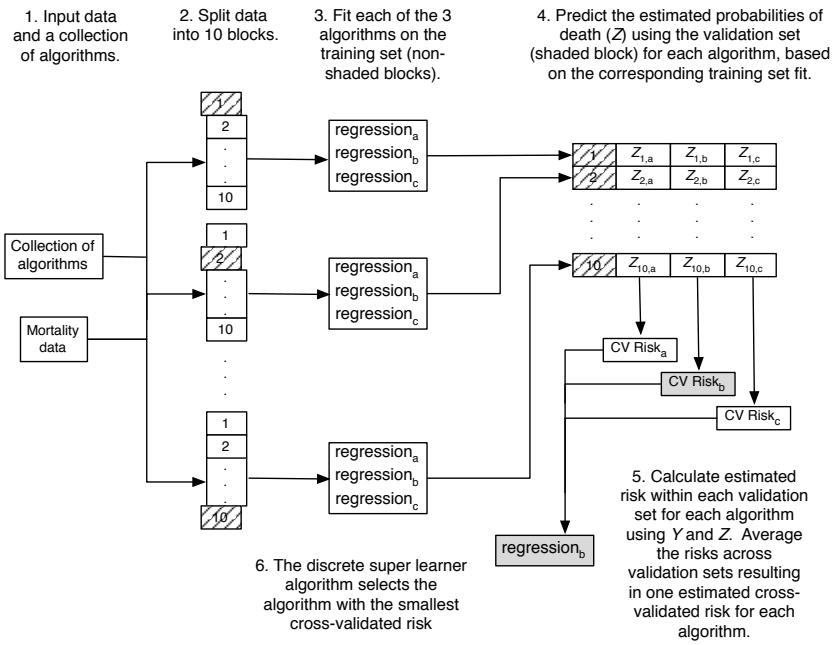

Fig. 3.1 Discrete super learner algorithm for the mortality study example where $\bar{Q}_n^b(A, W)$ is the algorithm with the smallest cross-validated risk

When the loss function is bounded, it has been shown that this discrete super learner will, for large sample sizes, perform as well as the algorithm that is the minimizer of the expected loss function. The latter impossible choice is called the oracle selector, which corresponds with simply selecting the estimator that is closest to the true $\bar{Q}_0$. In addition, cross-validation selection is tailored for small sample sizes, thus one should not be misled that cross-validation requires large sample sizes.

3.3.2 Super Learner

Can we improve upon the discrete super learner? Yes! We can use our three regressions to build a library of algorithms consisting of all weighted averages of these regressions. It is reasonable to expect that one of these weighted averages might perform better than one of the three regressions alone. This simple principle allows us to map a collection of candidate algorithms (in this case, our three regressions) into a library of weighted averages of these algorithms. Each weighted average is a unique candidate algorithm in this augmented library. We can then apply the same cross-validation selector to this augmented set of candidate algorithms, resulting in the super learner. It might seem that the implementation of such an estimator is problematic, since it requires minimizing the cross-validated risk over an infinite set of candidate algorithms (the weighted averages). The contrary is true. The super learner is not more computer intensive than the discrete super learner. If the discrete super learner has been implemented, then all the work has been done! Only the relatively trivial calculation of the optimal weight vector needs to be completed.

Consider that the discrete super learner has already been completed as described in Sect. 3.3.1. We then propose a family of weighted combinations of the three regression algorithms, which we index by the weight vector α. We want to determine which combination minimizes the cross-validated risk over the family of weighted combinations. The (cross-validated) probabilities of death (Z) for each algorithm are used as inputs in a working (statistical) model to predict the outcome Y. Therefore, we have a working model with three $\alpha = \{\alpha_a, \alpha_b, \alpha_c\}$ coefficients that need to be estimated, one for each of the three algorithms. Selecting the weights that minimize the cross-validated risk is a simple minimization problem, formulated as a regression of the outcomes Y on the predicted values of the algorithms (Z) according to the user-supplied parametric family of weighted combinations. The weighted combination with the smallest cross-validated risk is the "best" estimator according to our criterion: minimizing the estimated expected squared error loss function.

The selected weighted combination is a new estimator we can now use to input data (e.g., our complete mortality data set) to estimate predicted probabilities. Thus, we fit each of the three algorithms on our complete data (learning set). Combining these algorithm fits with our new estimator generates the super learner prediction function. This prediction function is the weighted combination of the candidate algorithms applied to the whole data set. See Fig. 3.2 for a full diagram of the super learner algorithm. In order to calculate an honest risk for the super learner, the super

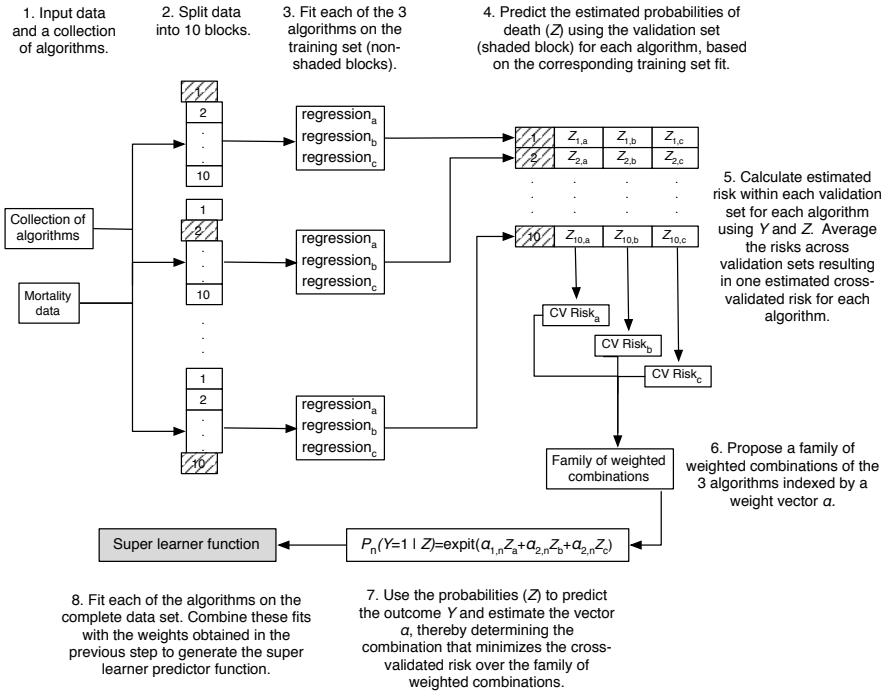

Fig. 3.2 Super learner algorithm for the mortality study example

learner itself must be externally cross-validated after the procedure described above has been implemented.

The family of weighted combinations includes only those α-vectors that have a sum equal to one, and where each weight is positive or zero. Theory does not dictate any restrictions on the family of weighted combinations used for assembling the algorithms; however, the restriction of the parameter space for α to be the convex combination of the algorithms provides greater stability for the final super learner prediction. The convex combination is not only empirically motivated, but also supported by theory. The oracle results for the super learner require a bounded loss function. Restricting oneself to a convex combination of algorithms implies that if each algorithm in the library is bounded, the convex combination will also be bounded.

The super learner improves asymptotically on the discrete super learner by working with a larger library. We reiterate that asymptotic results prove that in realistic scenarios (where none of the algorithms are a correctly specified parametric model), the cross-validated selector performs asymptotically as well as the oracle, which we define as the best estimator given the algorithms in the collection of algorithms. Consequently, the super learner performs asymptotically as well as the best choice among the family of weighted combinations of estimators. Thus, by adding more

competitors, we only improve the performance of the super learner. The asymptotic equivalence remains true if the number of algorithms in the library grows very quickly with sample size. Even when the collection of algorithms contains a correctly specified parametric statistical model, the super learner will approximate the truth as fast as the parametric statistical model, although it will be more variable.

The super learner algorithm provides a system to combine multiple estimators into an improved estimator, and returns a function we can also use for prediction in new data sets.

3.3.3 Finite Sample Performance and Applications

To examine the finite sample performance of the super learner we present a series of simulations and data applications. (For those readers unfamiliar with simulation, simulated data are ideal for methodology validation, as the true underlying distribution of the data is known.) We then demonstrate the super learner on a collection of real data sets and a microarray cancer data set.

Four different simulations are presented in this section. All four simulations involve a univariate X drawn from a uniform distribution in $[-4, 4]$. The outcomes follow the functions described below:

$$\begin{aligned}
&\textbf{Simulation 1: } Y = -2 \times \mathrm{I}(X < -3) + 2.55 \times \mathrm{I}(X > -2) - 2 \times \mathrm{I}(X > 0) \\
&\qquad\qquad + 4 \times \mathrm{I}(X > 2) - 1 \times \mathrm{I}(X > 3) + U; \\
&\textbf{Simulation 2: } Y = 6 + 0.4X - 0.36X^2 + 0.005X^3 + U; \\
&\textbf{Simulation 3: } Y = 2.83 \times \sin\left(\frac{\pi}{2} \times X\right) + U; \\
&\textbf{Simulation 4: } Y = 4 \times \sin(3\pi \times X) \times \mathrm{I}(X > 0) + U,
\end{aligned}$$

where $\mathrm{I}(\cdot)$ is the usual indicator function and U, our exogenous background error, is drawn from an independent standard normal distribution in all simulations. A sample of size 100 was drawn for each scenario. Figure 3.3 contains a scatterplot with a sample from each of the four simulations. The true curve for each simulation is represented by the solid line. These four simulations were chosen because they represent a diverse set of true regression functions, but all four have the same optimal $R^2 = 0.80$. The empirical R^2 is computed as $R^2 = 1 - (\sum(Y_i - Y_{i,n})^2 / \sum(Y_i - \bar{Y})^2)$, where $\bar{Y} = 1/n \sum_{i=1}^{n} Y_i$ and $Y_{i,n}$ is the predicted value of Y_i reported by the algorithm when applied to the whole data set. The optimal R^2 is the value attained when the true regression function (i.e., true conditional mean) is used and an infinite test sample is used to evaluate the mean squared errors in the numerator and denominator. Knowledge of the true regression function and using an infinite test sample implies $\sum(Y_i - Y_{i,n})^2 = \mathrm{var}(U) \times n = 1 \times n$. Hence the optimal R^2 in all four simulations

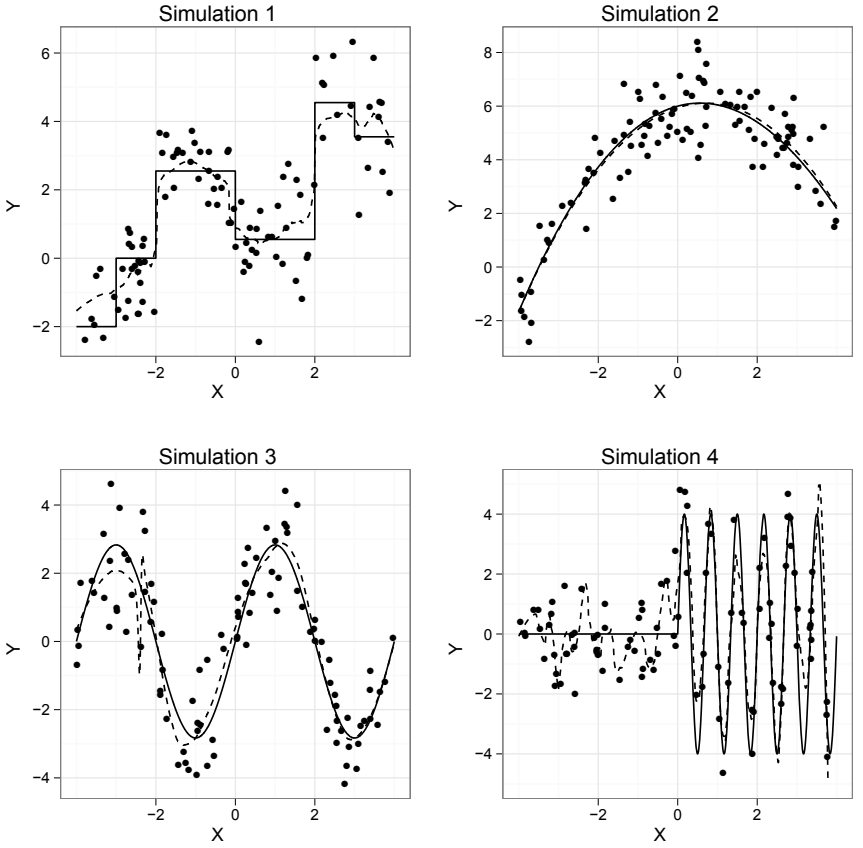

Fig. 3.3 Scatterplots of the four simulations. The *solid line* is the true relationship. The *points* represent one of the simulated data sets of size $n = 100$. The *dashed line* is the super learner fit for the shown data set

is $R^2_{opt} = 1 - (1/\text{var}(Y))$. The variance of Y is set such that $R^2_{opt} = 0.80$ in each simulation.

The collection of algorithms should ideally be a diverse set. One common aspect of many prediction algorithms is the need to specify values for tuning parameters. For example, generalized additive models require a degrees-of-freedom value for the spline functions and the neural network requires a size value. The tuning parameters could be selected using cross-validation or bootstrapping, but the different values of the tuning parameters could also be considered different prediction algorithms. A collection of algorithms could contain three generalized additive models with degrees of freedom equal to 2, 3, and 4. When one considers different values of tuning parameters as unique prediction algorithms in the collection, it is easy to see how the number of algorithms in the collection can become large.

Table 3.1 Collection of prediction algorithms for the simulations and citations

R Algorithm	Description	Source
glm	Linear model	R Development Core Team (2010)
interaction	Polynomial linear model	R Development Core Team (2010)
randomForest	Random forest	Liaw and Wiener (2002) Breiman (2001b)
bagging	Bootstrap aggregation of trees	Peters and Hothorn (2009) Breiman (1996d)
gam	Generalized additive models	Hastie (1992) Hastie and Tibshirani (1990)
gbm	Gradient boosting	Ridgeway (2007) Friedman (2001)
nnet	Neural network	Venables and Ripley (2002)
polymars	Polynomial spline regression	Kooperberg (2009) Friedman (1991)
bart	Bayesian additive regression trees	Chipman and McCulloch (2009) Chipman et al. (2010)
loess	Local polynomial regression	Cleveland et al. (1992)

In all four simulations, we started with the same collection of 21 prediction algorithms. Table 3.1 contains a list of the algorithms in the library. A linear model and a linear model with a quadratic term were considered. The default random forest algorithm, along with a collection of bagging regression trees with values of the complexity parameter (cp) equal to 0.10, 0.01, and 0.00 and a bagging algorithm adjusting the minimum split parameter to be 5, with default cp of 0.01, was also within the collection of algorithms. Generalized additive models with degrees of freedom equal to 2, 3, and 4 were added along with the default gradient boosting model. Neural networks with sizes 2 through 5, the polymars algorithm, and the Bayesian additive regression trees were added. Finally, we considered the loess curve with spans equal to 0.75, 0.50, 0.25, and 0.10.

Figure 3.3 contains the super learner fit on a single simulated data set for each scenario. With the given collection of algorithms, the super learner is able to adapt to the underlying structure of the data-generating function. For each algorithm we evaluated the true R^2 on a test set of size 10,000. The optimal R^2 is the value attained with knowledge of the true regression function. This value gives us an upper bound on the possible R^2 for each algorithm.

To assess the performance of the super learner in comparison to each algorithm, we simulated 100 samples of size 100 and computed the R^2 for each fit of the true regression function. The results are presented in Table 3.2. Negative R^2 values indicate that the mean is a better predictor of Y than the algorithm. In the first simulation, the regression-tree-based methods perform the best. Bagging complete regression trees (cp = 0) has the largest R^2. In the second simulation, the best algorithm is the quadratic linear regression (SL.interaction). In both of these cases, the super learner is able to adapt to the underlying structure and has an average R^2 close to the best algorithm. The same trend is exhibited in simulations 3 and 4; the super learner method of combining algorithms does nearly as well as the individual best

Table 3.2 Results for four simulations. Average R^2 based on 100 simulations and the corresponding standard errors

Algorithm	Sim 1		Sim 2		Sim 3		Sim 4	
	R^2	SE(R^2)	R^2	SE(R^2)	R^2	SE(R^2)	R^2	SE(R^2)
Super learner	0.741	0.032	0.754	0.025	0.760	0.025	0.496	0.122
Discrete SL	0.729	0.079	0.758	0.029	0.757	0.055	0.509	0.132
SL.glm	0.422	0.012	0.189	0.016	0.107	0.016	−0.018	0.021
SL.interaction	0.428	0.016	0.769	0.011	0.100	0.020	−0.018	0.029
SL.randomForest	0.715	0.021	0.702	0.027	0.724	0.018	0.460	0.109
SL.bagging(0.01)	0.751	0.022	0.722	0.036	0.723	0.018	0.091	0.054
SL.bagging(0.1)	0.635	0.120	0.455	0.195	0.661	0.029	0.020	0.025
SL.bagging(0.0)	0.752	0.021	0.722	0.034	0.727	0.017	0.102	0.060
SL.bagging(ms5)	0.747	0.020	0.727	0.030	0.741	0.016	0.369	0.104
SL.gam(2)	0.489	0.013	0.649	0.026	0.213	0.029	−0.014	0.023
SL.gam(3)	0.535	0.033	0.748	0.024	0.412	0.037	−0.017	0.029
SL.gam(4)	0.586	0.027	0.759	0.020	0.555	0.022	−0.020	0.034
SL.gbm	0.717	0.035	0.694	0.038	0.679	0.022	0.063	0.040
SL.nnet(2)	0.476	0.235	0.591	0.245	0.283	0.285	−0.008	0.030
SL.nnet(3)	0.700	0.096	0.700	0.136	0.652	0.218	0.009	0.035
SL.nnet(4)	0.719	0.077	0.730	0.062	0.738	0.102	0.032	0.052
SL.nnet(5)	0.705	0.079	0.716	0.070	0.731	0.077	0.042	0.060
SL.polymars	0.704	0.033	0.733	0.032	0.745	0.034	0.003	0.040
SL.bart	0.740	0.015	0.737	0.027	0.764	0.014	0.077	0.034
SL.loess(0.75)	0.599	0.023	0.761	0.019	0.487	0.028	−0.023	0.033
SL.loess(0.50)	0.695	0.018	0.754	0.022	0.744	0.029	−0.033	0.038
SL.loess(0.25)	0.729	0.016	0.738	0.025	0.772	0.015	−0.076	0.068
SL.loess(0.1)	0.690	0.044	0.680	0.064	0.699	0.039	0.544	0.118

algorithm. Since the individual best algorithm is not known a priori, if a researcher selected a single algorithm, they may do well in some data sets, but the overall performance will be worse than that of the super learner. For example, an individual who always uses bagging complete trees (SL.bagging(0.0)) will do well on the first three simulations, but will perform poorly on the fourth simulation compared to the average performance of the super learner.

In the first three simulations the super learner approaches the optimal R^2 value because algorithms in the collection approximate the truth well. However, in the fourth simulation, the collection is not rich enough to contain a combination of algorithms that approaches the optimal value. The super learner does as well as the best algorithms in the library but does not attain the optimal R^2. Upon supplementation of the collection of algorithms, the super learner achieves an average $R^2 = 0.76$, which is close to the optimal R^2 (results not shown; see Polley and van der Laan 2010).

To study the super learner in real data examples, we collected a number of publicly available data sets. Table 3.3 contains descriptions of the data sets, which can be found either in public repositories or in textbooks, with the corresponding citation listed in the table. Sample sizes ranged from 200 to 654 observations, and the number of covariates ranged from 3 to 18. All 13 data sets have a continuous outcome and no missing values. The collection of prediction algorithms included

Table 3.3 Description of data sets, where n is the sample size and p is the number of covariates

Name	n	p	Source
ais	202	10	Cook and Weisberg (1994)
diamond	308	17	Chu (2001)
cps78	550	18	Berndt (1991)
cps85	534	17	Berndt (1991)
cpu	209	6	Kibler et al. (1989)
FEV	654	4	Rosner (1999)
Pima	392	7	Newman et al. (1998)
laheart	200	10	Afifi and Azen (1979)
mussels	201	3	Cook (1998)
enroll	258	6	Liu and Stengos (1999)
fat	252	14	Penrose et al. (1985)
diabetes	366	15	Harrell (2001)
house	506	13	Newman et al. (1998)

the applicable algorithms from the univariate simulations along with the algorithms listed in Table 3.4. These algorithms represent a diverse set and should allow the super learner to work well in most practical settings. For comparison across data sets, we kept the collection of algorithms fixed for all data analyses.

In order to compare the performance of the K prediction algorithms across diverse data sets with outcomes on different scales, we used the relative mean squared error, which we denote RE for relative efficiency. The denominator is the mean squared error of a linear model:

$$\mathrm{RE}(k) = \frac{\mathrm{MSE}(k)}{\mathrm{MSE}(lm)}, \quad k = 1, \ldots, K.$$

The results for the super learner, the discrete super learner, and each individual algorithm can be found in Fig. 3.4. Each point represents the 10-fold cross-validated relative mean squared error for a data set, and the plus sign is the geometric mean of the algorithm across all 13 data sets. The super learner outperformed the discrete super learner, and both outperformed any individual algorithm. With real data, it is unlikely that one single algorithm would contain the true relationship, and the benefit of the combination of the algorithms vs. the selection of a single algorithm is demonstrated. The additional estimation of the combination parameters (α) does not cause an overfit in terms of the risk assessment. Among the individual algorithms, the Bayesian additive regression trees perform the best, but they overfit one of the data sets with a relative mean squared error of almost 3.0.

A common application of prediction is in microarray data. Super learning is well suited for this setting. Microarray data are often high dimensional, i.e., the number of covariates is larger than the sample size. We demonstrate the super learner in microarray data using a publicly available breast cancer data set published in van't Veer et al. (2002). This study was conducted to develop a gene-expression-based predictor for 5-year distant metastases. The outcome is a binary indicator that a

Table 3.4 Additional prediction algorithms in the collection of algorithms for the real data examples to be combined with the algorithms from Table 3.1

R Algorithm	Description	Source
bayesglm	Bayesian linear model	Gelman et al. (2010) Gelman et al. (2009)
glmnet	Elastic net	Friedman et al. (2010a) Friedman et al. (2010b)
DSA	DSA algorithm	Neugebauer and Bullard (2009) Sinisi and van der Laan (2004)
step	Stepwise regression	Venables and Ripley (2002)
ridge	Ridge regression	Venables and Ripley (2002)
svm	Support vector machine	Dimitriadou et al. (2009) Chang and Lin (2001)

Fig. 3.4 Tenfold cross-validated relative mean squared error compared to glm across 13 real data sets. Sorted by geometric mean, denoted by the plus (+) sign

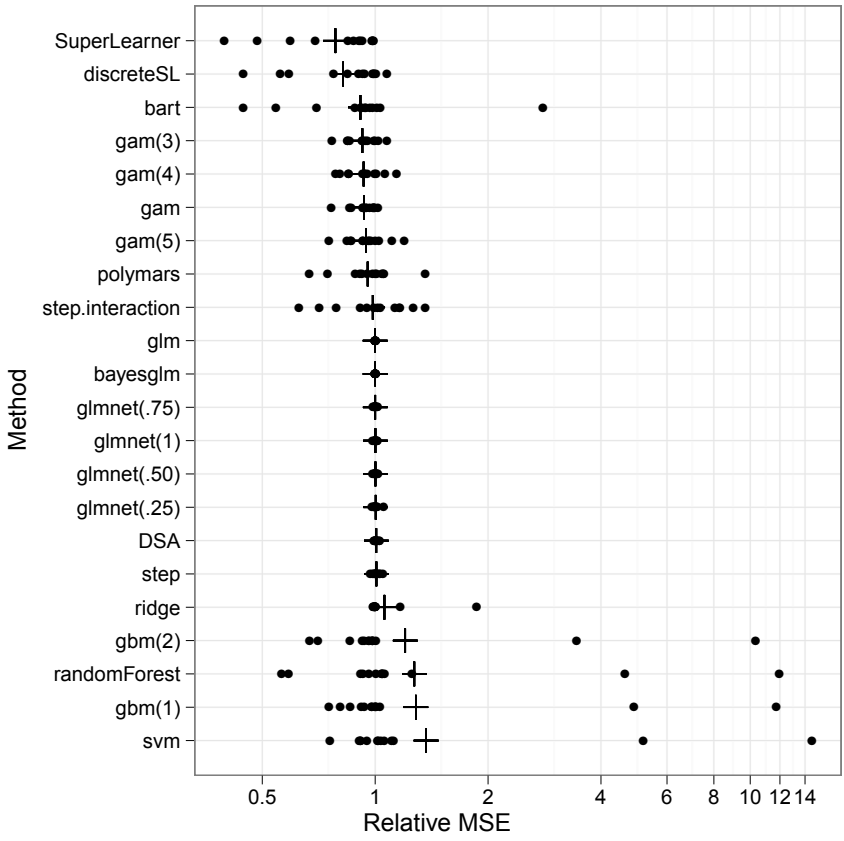

Table 3.5 Twentyfold cross-validated mean squared error for each algorithm and the standard error in the breast cancer study

Algorithm	Subset	Risk	SE
Super learner	–	0.194	0.0168
Discrete SL	–	0.238	0.0239
SL.knn(10)	All	0.249	0.0196
SL.knn(10)	Clinical	0.239	0.0188
SL.knn(10)	cor($p < 0.1$)	0.262	0.0232
SL.knn(10)	cor($p < 0.01$)	0.224	0.0205
SL.knn(10)	glmnet	0.219	0.0277
SL.knn(20)	All	0.242	0.0129
SL.knn(20)	Clinical	0.236	0.0123
SL.knn(20)	cor($p < 0.1$)	0.233	0.0168
SL.knn(20)	cor($p < 0.01$)	0.206	0.0176
SL.knn(20)	glmnet	0.217	0.0257
SL.knn(30)	All	0.239	0.0128
SL.knn(30)	Clinical	0.236	0.0119
SL.knn(30)	cor($p < 0.1$)	0.232	0.0139
SL.knn(30)	cor($p < 0.01$)	0.215	0.0165
SL.knn(30)	glmnet	0.210	0.0231
SL.knn(40)	All	0.240	0.0111
SL.knn(40)	Clinical	0.238	0.0105
SL.knn(40)	cor($p < 0.1$)	0.236	0.0118
SL.knn(40)	cor($p < 0.01$)	0.219	0.0151
SL.knn(40)	glmnet	0.211	0.0208
SL.glmnet(1.0)	cor(Rank = 50)	0.229	0.0285
SL.glmnet(1.0)	cor(Rank = 20)	0.208	0.0260
SL.glmnet(0.75)	cor(Rank = 50)	0.221	0.0269
SL.glmnet(0.75)	cor(Rank = 20)	0.209	0.0258
SL.glmnet(0.50)	cor(Rank = 50)	0.226	0.0269
SL.glmnet(0.50)	cor(Rank = 20)	0.211	0.0256
SL.glmnet(0.25)	cor(Rank = 50)	0.230	0.0266
SL.glmnet(0.25)	cor(Rank = 20)	0.216	0.0252
SL.randomForest	Clinical	0.198	0.0186
SL.randomForest	cor($p < 0.01$)	0.204	0.0179
SL.randomForest	glmnet	0.220	0.0245
SL.bagging	Clinical	0.207	0.0160
SL.bagging	cor($p < 0.01$)	0.205	0.0184
SL.bagging	glmnet	0.206	0.0219
SL.bart	Clinical	0.202	0.0183
SL.bart	cor($p < 0.01$)	0.210	0.0207
SL.bart	glmnet	0.220	0.0275
SL.mean	All	0.224	0.1016

patient had a distant metastasis within 5 years of initial therapy. In addition to the expression data, six clinical variables were attained. The clinical information was age, tumor grade, tumor size, estrogen receptor status, progesterone receptor status, and angioinvasion. The array data contained 4348 genes after the unsupervised screening steps outlined in the original article. We used the entire sample of 97 individuals (combining the training and validation samples from the original article) to fit the super learner.

In high-dimensional data, it is often beneficial to screen the variables before running prediction algorithms. Screening is part of the algorithm and should thus also be included when calculating the cross-validated risk of an algorithm in the super learner. Screening algorithms can be coupled with prediction algorithms to create new algorithms in the library. For example, we may consider k-nearest neighbors using all features and k-nearest neighbors on the subset of only clinical variables. These two algorithms are considered unique algorithms. Another screening algorithm involves testing the pairwise correlations of each variable with the outcome and ranking the variables by the corresponding p-value. With the ranked list of variables, we consider the screening cutoffs as follows: variables with a p-value less than 0.1, variables with a p-value less than 0.01, variables in the bottom 20, and variables in the bottom 50. An additional screening algorithm involves running the glmnet algorithm and selecting the variables with nonzero coefficients.

The results for the breast cancer data can be found in Table 3.5. The algorithms in the collection are k-nearest neighbors with $k = \{10, 20, 30, 40\}$, elastic net with $\alpha = \{1.0, 0.75, 0.50, 0.25\}$, random forests, bagging, bart, and an algorithm that uses the mean value of the outcome as the predicted probability. We coupled these algorithms with the screening algorithms to produce the full list of 38 algorithms. Within this collection of algorithms, the best algorithm in terms of minimum risk estimate is the random forest algorithm using only the clinical variables (MSE = 0.198). As we observed in the previous examples, the super learner was able to attain a risk comparable to the best algorithm (MSE = 0.194).

3.4 Road Maps

In previous chapters, we introduced our road map for targeted learning (Fig. 3.5), the first steps of which involved defining our data, model, and target parameter. This chapter dealt with obtaining the best initial estimator of the relevant portion Q_0 of the distribution P_0 of O. The next stage of the road map addresses estimation of the target parameter using TMLE, taking this initial estimator as input.

We also present a separate road map for prediction estimation questions in Fig. 3.6. We note that inference for prediction is not covered in detail in this text and we refer readers in Sect. 3.7 to literature using the permutation distribution for obtaining exact tests of the null hypothesis of independence of the covariates and the outcome, allowing the incorporation of machine learning.

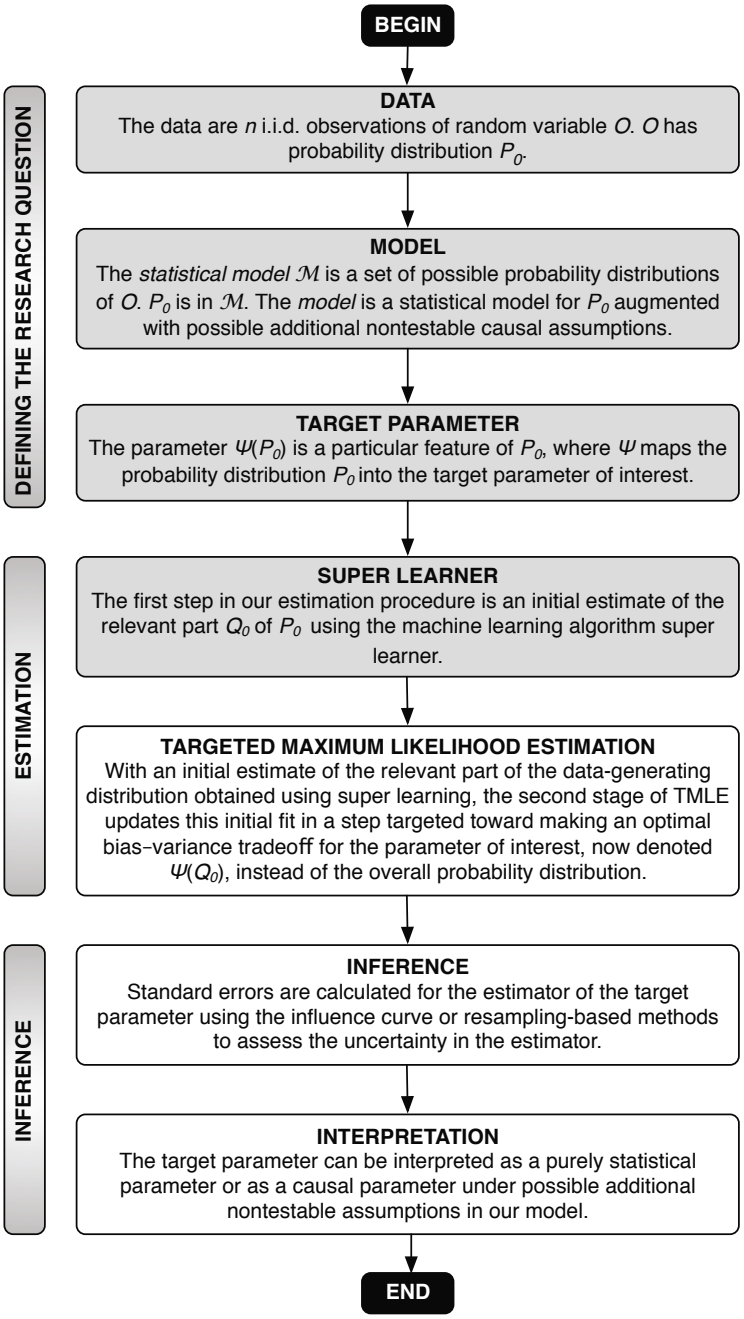

Fig. 3.5 Road map for targeted learning

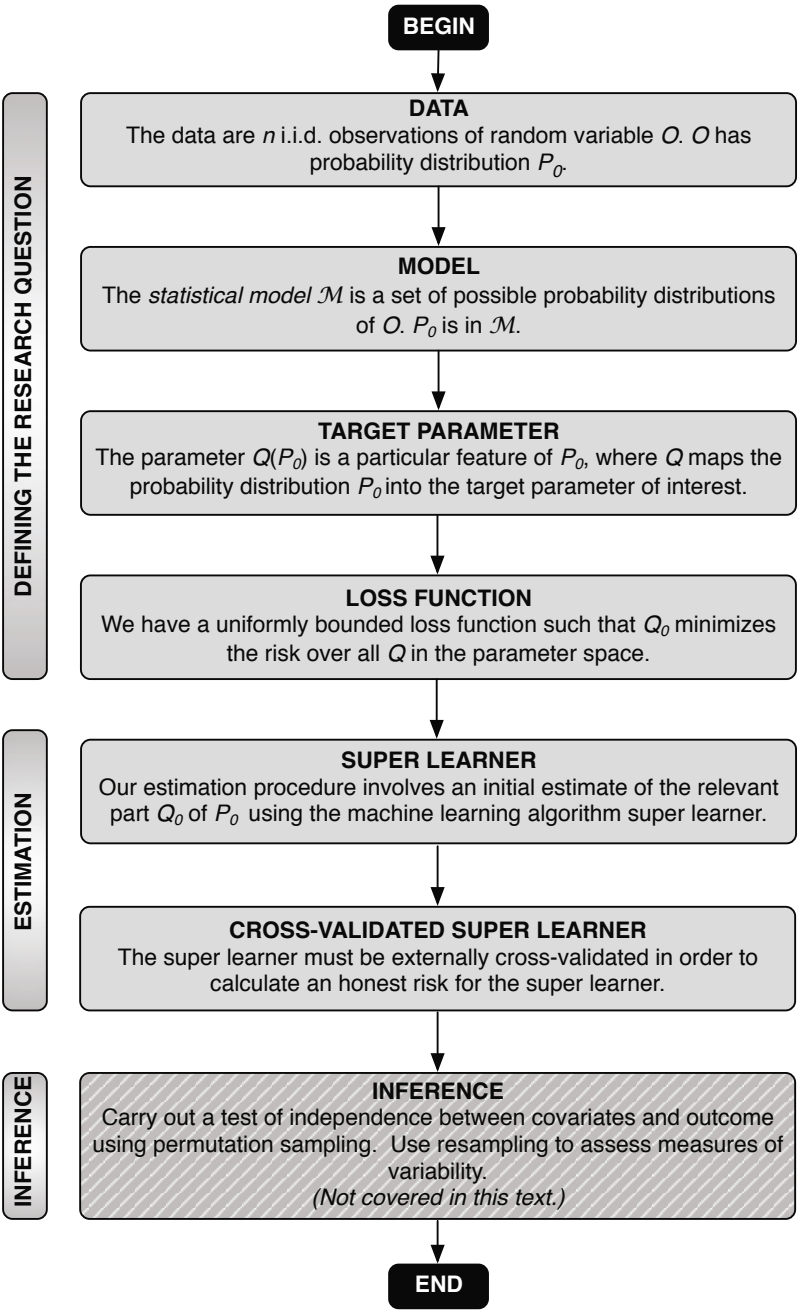

Fig. 3.6 Road map for prediction

3.5 Conceptual Framework of Loss-Based Super Learning

Suppose we observe n i.i.d. observations $O_1, \ldots, O_n$ on a random variable O from a probability distribution P_0 known to be an element of a statistical semiparametric statistical model $\mathcal{M}$. Our goal is to learn a particular parameter of P_0, which we will denote by $Q(P_0)$, and let $\mathcal{Q} = \{Q(P) : P \in \mathcal{M}\}$ be the parameter space. We assume that we have available a loss function $L(Q)(O)$ such that Q_0 minimizes the risk $P_0 L(Q) \equiv E_0 L(Q)(O)$ of Q over all Q in the parameter space $\mathcal{Q}$. In addition, it is assumed that this loss function is uniformly bounded so that $P_0(L(Q)(O) < M) = 1$ for some universal constant M, uniformly in all $Q \in \mathcal{Q}$. A library of candidate estimators of Q_0, the choice of loss function, and a choice of cross-validation scheme now define the super learner of Q_0.

Creating a better estimator from among the available estimators. If the parameter $Q : \mathcal{M} \to \mathcal{Q}$ is not pathwise differentiable (i.e., not identifiable and smooth enough to allow central-limit-theorem-based inference), then there is no efficiency theory. That is, even asymptotically, there is no best estimator of Q_0. As a consequence, the best one can do is to make sure that one is better than any competitor. This can be done by including any competing algorithm in the super learner collection of algorithms. Thus one has a large collection of candidate estimators. These candidate estimators should use the knowledge that $P_0 \in \mathcal{M}$. That is, each estimator should at a minimum map the data into functions in the parameter space $\mathcal{Q}$.

One particular approach might require a choice of a number of fine-tuning parameters. Such an approach would generate many members for the collection. An estimator could be combined with different dimension-reduction approaches, so that one estimator would result in several members in the collection. One might also partition the outcome space for O and stratify estimators accordingly, and also consider applying different estimators to different strata. In this manner, one estimation procedure and several stratification variables would map into a whole collection of estimators for the super learner library.

There is no point in painstakingly trying to decide which estimators to enter in the collection; instead add them all. The theory supports this approach, and finite sample simulations and data analyses only confirm that it is very hard to overfit the super learner by augmenting the collection, but benefits are obtained. Indeed, for large data sets, we simply do not have enough algorithms available to build the desired collection that would fully utilize the power of the super learning principle as established by the oracle result.

The free lunch: fully robust application-specific modeling. This is not enough. In a particular application, there are always many experts with creative ideas. Put them in a room and let them generate ideas about effective dimension reductions, propose parametric statistical models they find interesting, and let them propose strategies for approximating this unknown true target-function Q_0. Translate these into new candidate algorithms for the collection of algorithms. For example, one professor might think that certain specific summary measures of the history of the unit at

baseline should be particularly effective in predicting the outcome of interest. This then translates into estimators that only use these summaries and algorithms that add these summary measures to the existing set of (nontransformed) variables. If there are competing theories about how the truth is best approximated, translate them all into candidate estimators for the super learning library. The super learner will not select an estimator that performs poorly, but even mediocre algorithms can still improve the super learner. That is, there is no risk in adding candidate estimators that are heavily model-based to the super learner; there is only benefit.

Concerned about overfitting the super learner? Indeed, let the data speak to answer this question. That is why one should evaluate the performance of the super learner itself by determining its cross-validated risk. It can then be determined if the super learner does as well or better than any of the candidate algorithms in the collection. In particular, one might diagnose that one has reached a point at which adding more algorithms harms the super learner performance, but our experience has not reached that point by any meaningful standard. If anything, it appears to flatten out, but not deteriorate. However, it is important that one use quite high-fold cross-validation when evaluating the super learner itself (say 20-fold), especially when the sample size gets small. Above all, it is crucial that the family of combinations respects a universal bound on the loss functions across all combinations.

Computational challenge. The super learning system of learning is perfectly tailored for parallel programming. The different candidate estimators can do their job separately, and the applications of the candidate estimators to the different training sets can be separated as well.

Generality of super learning. Super learning can be applied to estimate an immense class of parameters across different data structures O and different statistical models $\mathcal{M}$. One can use it to estimate marginal densities, conditional densities, conditional hazards, conditional means, conditional medians, conditional quantiles, conditional survival functions, and so on, under biased sampling, missingness, and censoring.

It is a matter of defining Q_0 as a parameter of P_0 and determining an appropriate loss function. For example, the minus log loss function can be used for conditional densities and hazards. However, one might come up with loss functions that are indexed by unknown parameters: $L_h(Q)$ for some unknown h. Many such examples are now provided in the literature, such as the double robust augmented inverse probability of treatment-weighted loss function for the treatment-specific mean outcome as a function of effect modifiers. In this case h includes a conditional distribution of treatment as a function of the covariates. This nuisance parameter would be known in an RCT, but it will need to be estimated in an observational study. We discuss the statistical property of double robustness in Chap. 6.

In these situations one will need to estimate h from the data and then employ this estimated loss function as before. The basic messsage is that one needs the estimation of h to be easier than the estimation of Q_0 in order to get the full benefit

of super learner, as if h was known from the start. It should also be remarked that h could represent an index that does not affect the validity of the loss function in the sense that for each choice of h, Q_0 minimizes the risk of $L_h(Q)$ over all Q. In these cases, the choice of h only affects the dissimilarity measure for which the performance of the super learner is optimized. For example, h might represent a weight function in a squared-error loss function, or it might represent a covariance matrix for the generalized squared-error loss function for a conditional mean $E(\mathbf{Y} \mid W)$ of a multivariate outcome:

$$L_h(Q)(W, \mathbf{Y}) = (\mathbf{Y} - Q(W))^\top h(W)(\mathbf{Y} - Q(W)).$$

Choice of loss function. If several choices are available, the loss function that maps into the desired dissimilarity measure $d_L(Q, Q_0) = E_0 L(Q) - E_0 L(Q_0)$ should be selected. It should be kept in mind that the super learner optimizes the approximation of Q_0 with respect to this dissimilarity d_L implied by the loss function. For example, suppose one wishes to estimate a conditional survival function at a time point t_0, $Q_0 = P(T > t_0 \mid W)$. Then one could still use the minus log loss function for the conditional density of T, given W, which, by substitution, also implies a valid loss function for Q_0, since Q_0 is determined by this conditional density. However, this loss function is trying to determine an entire conditional density, and is thus not very targeted towards its goal. Instead, we can use

$$L(Q)(W, T) = [I(T > t_0) - Q(W)]^2 .$$

This loss function is minimized over all functions Q of W by $Q_0 = P(T > t_0 \mid W)$, and thereby targets exactly our parameter of interest. Indeed, the super learner will now be aiming to minimize the dissimilarity:

$$d_L(Q, Q_0) = E_0[Q(W) - Q_0(W)]^2,$$

i.e., the expected squared error between the candidate survival function at t_0 and the true survival function at t_0.

Formal oracle result for cross-validation selector. Consider a loss function that satisfies

$$\sup_Q \frac{\mathrm{var}_{P_0}\{L(Q) - L(Q_0)\}}{P_0\{L(Q) - L(Q_0)\}} \leq M_2 \tag{3.1}$$

and that is uniformly bounded:

$$\sup_{O,Q} \mid L(Q) - L(Q_0) \mid (O) < M_1 < \infty,$$

where the supremum is over the support of P_0 and over all possible candidate estimators of Q_0 that will ever be considered. We used the notation $P_0 f = \int f(o) dP_0(o)$ for the expectation of $f(O)$ under P_0. The first property (3.1) applies to the log-likelihood loss function and any weighted squared residual loss function, among

others. Property (3.1) is essentially equivalent to the assumption that the loss-function-based dissimilarity $d(Q, Q_0) = P_0\{L(Q) - L(Q_0)\}$ is quadratic in a distance between Q and Q_0. Property (3.1) has been proven for log-likelihood loss functions and weighted L^2-loss functions and is in essence equivalent to stating that the loss function implies a quadratic dissimilarity $d(Q, Q_0)$ (van der Laan and Dudoit 2003). If this property does not hold for the loss function, the rates $1/n$ for second-order terms in the below stated oracle inequality reduce to the rate $1/\sqrt{n}$.

Let $B_n \in \{0, 1\}^n$ be a random variable that splits the learning sample in a training sample $\{i : B_n(i) = 0\}$ and validation sample $\{i : B_n(i) = 1\}$, and let P^0_{n,B_n} and P^1_{n,B_n} denote the empirical distribution of the training and validation sample, respectively. Given candidate estimators $P_n \to \hat{Q}_k(P_n)$, the loss-function-based cross-validation selector is now defined by

$$k_n = \hat{K}(P_n) = \arg\min_k E_{B_n} P^1_{n,B_n} L(\hat{Q}_k(P^0_{n,B_n})).$$

The resulting estimator, the discrete super learner, is given by $\hat{Q}(P_n) = \hat{Q}_{\hat{K}(P_n)}(P_n)$.

For quadratic loss functions, the cross-validation selector satisfies the following (so-called) oracle inequality: for any $\delta > 0$

$$E_{B_n}\{P_0 L(\hat{Q}_{k_n}(P^0_{n,B_n}) - L(Q_0)\} \le (1 + 2\delta) E_{B_n} \min_k P_0\{L(\hat{Q}_k(P^0_{n,B_n})) - L(Q_0)\} + 2C(M_1, M_2, \delta)\frac{1 + \log K(n)}{np},$$

where the constant $C(M_1, M_2, \delta) = 2(1+\delta)^2(M_1/3+M_2/3)$ (van der Laan and Dudoit 2003, p. 25). This result proves [see van der Laan and Dudoit (2003) for the precise statement of these implications] that if the number of candidates $K(n)$ is polynomial in sample size, then the cross-validation selector is either asymptotically equivalent to the oracle selector (based on a sample of training sample sizes, as defined on the right-hand side of the above inequality), or it achieves the parametric rate $\log n/n$ for convergence with respect to $d(Q, Q_0) \equiv P_0\{L(Q) - L(Q_0)\}$.

So in most realistic scenarios, in which none of the candidate estimators achieves the rate of convergence one would have with an a priori correctly specified parametric statistical model, the cross-validated estimator selector performs asymptotically exactly as well (not only in rate, but also up to the constant!) as the oracle-selected estimator. These oracle results are generalized for estimated loss functions $L_n(Q)$ that approximate a fixed loss function $L(Q)$. If $\arg\min_Q P_0 L_n(Q) \neq Q_0$, then the oracle inequality also presents second-order terms due to the estimation of the loss function (van der Laan and Dudoit 2003).

3.6 Notes and Further Reading

We've discussed in this chapter the notion of estimator selection. We use this terminology over "model selection," since the formal meaning of a (statistical) model in

the field of statistics is the set of possible probability distributions, and most algorithms are not indexed by a statistical model choice. The general loss-based super learner was initially presented in van der Laan et al. (2007b). Super learner is a generalization of the stacking algorithm introduced in the neural networks context by Wolpert (1992) and adapted to the regression context by Breiman (1996c), and its name was introduced due to the theoretical oracle property and its consequences as presented in van der Laan and Dudoit (2003). The stacking algorithm is examined in LeBlanc and Tibshirani (1996) and the relationship to the model-mix algorithm of Stone (1974) and the predictive sample-reuse method of Geisser (1975) is discussed. Recent literature on aggregation and ensemble learners includes Tsybakov (2003), Juditsky et al. (2005), Bunea et al. (2006, 2007a,b), and Dalalyan and Tsybakov (2007, 2008). As noted previously, inference for prediction, such as permutation resampling, is not covered in this text. We refer the interested reader to Lehmann (1986), Hastie et al. (2001), Ruczinski et al. (2002), Birkner et al. (2005), and Chaffee et al. (2010). The simulations and data analyses contained in this chapter were previously published as a technical report (Polley and van der Laan 2010).

Chapter 15 uses super learning to estimate the risk score of mortality in a Kaiser Permanente database. Additionally, Chap. 16 discusses the use of super learning in right-censored data. We refer readers to Polley and van der Laan (2009) for a chapter in the book *Design and Analysis of Clinical Trials with Time-to-Event Endpoints* that discusses the use of super learning to assess effect modification in clinical trials.

Theory for loss-function-based cross-validation is presented in van der Laan and Dudoit (2003), including the finite sample oracle inequality, the asymptotic equivalence of the cross-validation selector, and the oracle selector. See also van der Laan et al. (2006), van der Vaart et al. (2006), van der Laan et al. (2004), Dudoit and van der Laan (2005), Keleş et al. (2002), and Sinisi and van der Laan (2004). A finite sample result for the single-split cross-validation selector for the squared error loss function was established in Györfi et al. (2002) and then generalized in van der Laan and Dudoit (2003) and Dudoit and van der Laan (2005) for both general cross-validation schemes and a general class of loss functions.

Other types of cross-validation beyond V-fold cross-validation include bootstrap cross-validation, Monte Carlo cross-validation, and leave-one-out cross-validation (Stone 1974, 1977; Breiman et al. 1984; Breiman and Spector 1992; Efron and Tibshirani 1993; Breiman 1996a,b; Ripley 1996; Breiman 1998; Hastie et al. 2001; Ambroise and McLachlan 2002; Györfi et al. 2002). Simulation studies (Pavlic and van der Laan 2003) show that likelihood-based cross-validation performs well when compared to common validity-functionals-based approaches, such as Akaike's information criterion (Akaike 1973; Bozdogan 2000), Bayesian Information criterion (Schwartz 1978), minimum description length (Rissanen 1978), and informational complexity (Bozdogan 1993).

Hastie et al. (2001) covers a variety of machine learning algorithms and related topics. Areas include stepwise selection procedures, ridge regression, LASSO, principal component regression, least angle regression, nearest neighbor methods, random forests, support vector machines, neural networks, classification methods, kernel smoothing methods, and ensemble learning.

Chapter 4
Introduction to TMLE

Sherri Rose, Mark J. van der Laan

This is the second chapter in our text to deal with estimation. We started by defining the research question. This included our data, model for the probability distribution that generated the data, and the target parameter of the probability distribution of the data. We then presented the estimation of prediction functions using super learning. This leads us to the estimation of causal effects using the TMLE. This chapter introduces TMLE, and a deeper understanding of this methodology is provided in Chap. 5. Note that we use the abbreviation *TMLE* for *targeted maximum likelihood estimation* and the *targeted maximum likelihood estimator*. Later in this text, we discuss *targeted minimum loss-based estimation*, which can also be abbreviated *TMLE*.

For the sake of demonstration, we have considered the data structure $O = (W, A, Y) \sim P_0$. Our statistical model for the probability distribution P_0 is nonparametric. The target parameter for this example is $E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, which can be interpreted as a causal effect under nontestable assumptions formalized by an SCM, including the randomization assumption and the positivity assumption. In Chap. 3, we estimated $E_0(Y \mid A, W)$ using super learning. With super learning we are able to respect that the statistical model does not allow us to assume a particular parametric form for the prediction function $E_0(Y \mid A, W)$. We could have estimated the entire conditional density of the outcome Y, but then we would be estimating portions of the density we do not need. In particular, this would mean that our initial estimator, such as a super learner of this conditional density of Y, would be targeted toward the complete conditional density, even though it is better to target it toward the conditional mean of Y. Estimating only the relevant portion of the density of O in this first step of the TMLE procedure provides us with a maximally efficient (precise) and unbiased procedure: the practical and asymptotic performance of the TMLE of ψ_0 only cares about how well $\bar{Q}_0$ is estimated.

The super learner fit can be plugged into the target parameter mapping to obtain a corresponding estimator of the target parameter. In other words, for each subject in the sample, one would evaluate the difference between the predicted value of Y under treatment ($A = 1$) and control ($A = 0$) and average these differences across all subjects in the sample.

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_4,

However, this super learner maximum likelihood (ML)-based substitution estimator is not targeted toward the parameter of interest. The super learner prediction function was tailored to optimally fit the overall prediction function $E_0(Y \mid A, W)$, spreading its errors uniformly to (successfully) optimize average squared prediction errors, and thereby suffers from a nonoptimal bias–variance tradeoff for the causal effect of interest. Specifically, this ML-based super learner of the causal effect will be biased.

Our TMLE procedure improves on the ML-based substitution estimator by reducing bias for the target parameter of interest. The initial super learner fit for $E_0(Y \mid A, W)$ is the first step in the TMLE procedure. The second stage of the TMLE procedure is a step targeted toward modifying the initial estimator of $E_0(Y \mid A, W)$ in order to make it less biased for the target parameter. That is, the second stage of TMLE is tailored to get the best estimate of our target parameter of interest, with respect to bias and variance, instead of a best estimate of the overall prediction function $E_0(Y \mid A, W)$. We cover the entire TMLE procedure in this chapter, assuming the reader has knowledge based on the material presented in Chap. 3.

We explain the TMLE procedure in multiple ways in these two chapters, with the goal of reinforcing the method and targeting different levels of understanding (conceptual, applied, theoretical). Thus, the applied researcher may only be interested in a thorough understanding of the conceptual and applied sections, whereas the more theoretically inclined mathematician may wish to also read the technical derivations and Appendix A.

TMLE Methodology Summary

TMLE is a two-step procedure where one first obtains an estimate of the data-generating distribution P_0, or the relevant portion Q_0 of P_0. The second stage updates this initial fit in a step targeted toward making an optimal bias–variance tradeoff for the parameter of interest $\Psi(Q_0)$, instead of the overall density P_0. The procedure is double robust and can incorporate data-adaptive likelihood-based estimation procedures to estimate Q_0 and the treatment mechanism. The double robustness of TMLE has important implications in both randomized controlled trials and observational studies, with potential reductions in bias and gains in efficiency.

We use our mortality study example to present an application of TMLE. As a reminder, in this study we are interested in the effect of LTPA on death. We have binary Y, death within 5 years of baseline, and binary A indicating whether the subject meets recommended levels of physical activity. The data structure in this example is $O = (W, A, Y) \sim P_0$. While we use this basic data structure and a particular target parameter to illustrate the procedure, TMLE is a very flexible general method for estimating any particular target parameter of a true probability distribution that is known to be an element of any particular statistical model. We will demonstrate its implementation with a variety of specific data structures throughout this text. In Appendix A, we also present a general TMLE of causal effects of

multiple time point interventions for complex longitudinal data structures. However, we find introducing TMLE in the context of a simple data structure is helpful for many people. Starting with Appendix A is often overwhelming, and that appendix is geared toward those who desire a comprehensive and rigorous statistical understanding or wish to develop TMLE for unique applications encountered in practice, corresponding with a choice of data structure, statistical model, and target parameter, not previously addressed.

TMLE has many attractive properties that make it preferable to other existing estimators of a target parameter of the probability distribution of the data. We fully detail these properties in Chaps. 5 and 6, after introducing them in this chapter, and compare other estimators to TMLE based on these properties. Of note, TMLE removes all the asymptotic residual bias of the initial estimator for the target parameter, if it uses a consistent estimator of the treatment mechanism. If the initial estimator was already consistent for the target parameter, the slight additional fitting of the data in the targeted step will potentially remove some finite sample bias, and certainly preserve this consistency property of the initial estimator.

As a consequence, the TMLE is a so-called double robust estimator. In addition, if the initial estimator and the estimator of the treatment mechanism are both consistent, then it is also asymptotically efficient according to semiparametric statistical model efficiency theory. It allows the incorporation of machine learning (i.e., super learning) methods for the estimation of both $\bar{Q}_0$ and g_0 so that we do not make assumptions about the probability distribution P_0 we do not believe. In this manner, every effort is made to achieve minimal bias and the asymptotic semiparametric efficiency bound for the variance.

TMLE is also a substitution estimator. Substitution estimators are plug-in estimators, taking an estimator of the relevant part of the data-generating distribution and plugging it into the mapping $\Psi()$. Substitution estimators respect the statistical model space (i.e., the global constraints of the statistical model) and respect that the target parameter ψ_0 is a number obtained by applying the target parameter mapping Ψ to a particular probability distribution in the statistical model. Substitution estimators are therefore more robust to outliers and sparsity than nonsubstitution estimators.

4.1 Motivation

Let us step back for a moment and discuss why we are here. We want to estimate a parameter $\Psi(P_0)$ under a semiparametric statistical model that represents actual knowledge. Thus we don't want to use a misspecified parametric statistical model that makes assumptions we know to be false. We also know that an ML-based substitution estimator is not targeted to the parameter we care about. While we like this approach as it is flexible, it is still not a targeted approach. TMLE is a *targeted* substitution estimator that incorporates super learning to get the best estimate of our

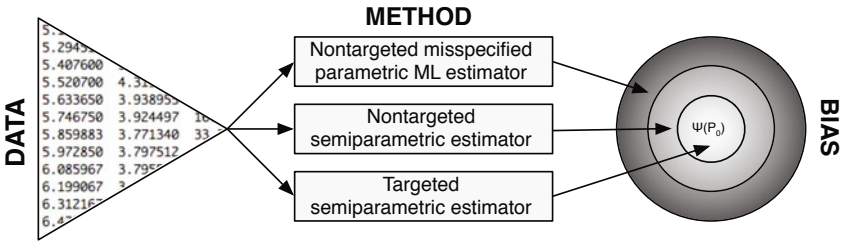

Fig. 4.1 Illustration of bias for different methods

target parameter; it is tailored to be a minimally biased method while also being tailored to fully utilize all the information in the data.

We illustrate this in Fig. 4.1. The outermost ring is furthest from the truth, and that represents the estimate we achieve using a misspecified parametric statistical model. The middle ring in our target improves on the misspecified parametric statistical model, but it still does not contain the truth. This ring is our nontargeted semiparametric statistical model approach (super learning). The innermost circle contains the true $\Psi(P_0)$, and this is what we have the potential to achieve with super learning *and* TMLE combined. We refer to the combined two-stage approach as TMLE, even though it is understood that the initial estimator and estimator of the treatment mechanism should be based on super learning respecting the actual knowledge about P_0.

4.2 TMLE in Action: Mortality Study Example

In Chap. 3, we discussed the implementation of super learning for our simplified mortality study example. In this section we analyze the actual data, updating the super learner estimate of $\bar{Q}_0$ with a targeting step. This section serves as an introduction to the implementation of TMLE in a concrete example: the data structure is $O = (W, A, Y) \sim P_0$, the nonparametric statistical model is augmented with causal assumptions, and the targeted parameter is $\Psi(P_0) = E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, which represents the causal risk difference under these causal assumptions. The mean over the covariate vector W in $\Psi(P_0)$ is simply estimated with the empirical mean, so that our substitution TMLE will be of the type

$$\psi_n = \Psi(Q_n) = \frac{1}{n}\sum_{i=1}^{n}\{\bar{Q}_n(1, W_i) - \bar{Q}_n(0, W_i)\},$$

where $Q_n = (\bar{Q}_n, Q_{W,n})$ and $Q_{W,n}$ is the empirical distribution for the marginal distribution of W. The second step in the TMLE will update our initial estimate of $\bar{Q}_0$. We will use the superscript 0 to denote this initial estimate, in conjunction with the

Table 4.1 SPPARCS variables

Variable	Description
Y	Death occurring within 5 years of baseline
A	LTPA score $\geq$ 22.5 METs at baseline[‡]
W_1	Health self-rated as "excellent"
W_2	Health self-rated as "fair"
W_3	Health self-rated as "poor"
W_4	Current smoker
W_5	Former smoker
W_6	Cardiac event prior to baseline
W_7	Chronic health condition at baseline
W_8	$x \leq 60$ years old
W_9	$60 < x \leq 70$ years old
W_{10}	$80 < x \leq 90$ years old
W_{11}	$x > 90$ years old
W_{12}	Female

[‡] LTPA is calculated from answers to a detailed questionnaire where prior performed vigorous physical activities are assigned standardized intensity values in metabolic equivalents (METs). The recommended level of energy expenditure for the elderly is 22.5 METs.

subscript n thus we have $\bar{Q}_n^0$ as our initial estimate of $\bar{Q}_0$. Information from the treatment mechanism (or exposure mechanism; we use these terms interchangeably) is used to update $\bar{Q}_n^0$ and target it toward the parameter of interest. In this example, our treatment mechanism is $g_0 = P_0(A \mid W)$. Our updated estimate of $\bar{Q}_0$ is denoted $\bar{Q}_n^1$.

Data. The National Institute of Aging-funded Study of Physical Performance and Age-Related Changes in Sonomans (SPPARCS) is a population-based, census-sampled, study of the epidemiology of aging and health. Participants of this longitudinal cohort were recruited if they were aged 54 years and over and were residents of Sonoma, CA or surrounding areas. Study recruitment of 2092 persons occurred between May 1993 and December 1994 and follow-up continued for approx. 10 years. The data structure is $O = (W, A, Y)$, where $Y = I(T \leq 5 \text{ years})$, T is time to the event death, A is a binary categorization of LTPA, and W are potential confounders. These variables are further defined in Table 4.1. Of note is the lack of any right censoring in this cohort. The outcome (death within or at 5 years after baseline interview) and date of death was recorded for each subject. Our parameter of interest is the causal risk difference, the average treatment effect of LTPA on mortality 5 years after baseline interview. The cohort was reduced to a size of $n = 2066$, as 26 subjects were missing LTPA values or self-rated health score (1.2% missing data).

4.2.1 Estimator

Estimating $\bar{Q}_0$. In Chap. 3, we generated a super learner prediction function. This is the first step in our TMLE procedure. Thus, we take as inputs our super learner

Table 4.2 Collection of algorithms

Algorithm	Description
glm	Linear model
bayesglm	Bayesian linear model
polymars	Polynomial spline regression
randomForest	Random forest
glmnet,$\alpha = 0.25$	Elastic net
glmnet,$\alpha = 0.50$	
glmnet,$\alpha = 0.75$	
glmnet,$\alpha = 1.00$	
gam, degree = 2	Generalized additive models
gam, degree = 3	
gam, degree = 4	
gam, degree = 5	
nnet,size = 2	Neural network
nnet, size = 4	
gbm, interaction depth=1	Gradient boosting
gbm, interaction depth=2	

prediction function, the initial estimate $\bar{Q}_n^0$, and our data matrix. The data matrix includes columns for each of the covariates W found in Table 4.1, exposure LTPA (A), and outcome Y indicating death within 5 years of baseline. This is step 1 as described in Fig. 4.2. We implemented super learner in the R programming language (R Development Core Team 2010), using the 16 algorithms listed in Table 4.2, recalling that algorithms of the same class with different tuning parameters are considered individual algorithms. Then we calculated predicted values for each of the 2066 observations in our data set, using their observed value of A, and added this as an n-dimensional column labeled $\bar{Q}_n^0(A_i, W_i)$ in our data matrix. Then we calculated a predicted value for each observation where we set $a = 1$, and also $a = 0$, forming two additional columns $\bar{Q}_n^0(1, W_i)$ and $\bar{Q}_n^0(0, W_i)$. Note that for those observations with an observed value of $A_i = 1$, the value in column $\bar{Q}_n^0(A_i, W_i)$ will be equal to the value in column $\bar{Q}_n^0(1, W_i)$. For those with observed $A_i = 0$, the value in column $\bar{Q}_n^0(A_i, W_i)$ will be equal to the value in column in $\bar{Q}_n^0(0, W_i)$. This is depicted in step 2 of Fig. 4.2. At this stage we could plug our estimates $\bar{Q}_n^0(1, W_i)$ and $\bar{Q}_n^0(0, W_i)$ for each subject into our substitution estimator of the risk difference:

$$\psi_{MLE,n} = \Psi(Q_n) = \frac{1}{n} \sum_{i=1}^{n} \{\bar{Q}_n^0(1, W_i) - \bar{Q}_n^0(0, W_i)\}.$$

This is the super learner ML-based substitution estimator discussed previously, plugging in the empirical distribution $Q_{W,n}^0$ for the marginal distribution of W, and the super learner $\bar{Q}_n^0$ for the true regression $\bar{Q}_0$. We know that this estimator is not targeted towards the parameter of interest, so we continue on to a targeting step.

Estimating g_0. Our targeting step required an estimate of the conditional distribution of LTPA given covariates W. This estimate of $P_0(A \mid W) \equiv g_0$ is denoted g_n and was obtained using super learning and the same algorithms listed in Table 4.2. We estimated predicted values using this new super learner prediction function, adding two more columns to our data matrix: $g_n(1 \mid W_i)$ and $g_n(0 \mid W_i)$. This can be seen in Fig. 4.2 as step 3.

Determining a parametric working model to fluctuate the initial estimator. The targeting step used the estimate g_n in a clever covariate to define a parametric working model coding fluctuations of the initial estimator. This clever covariate $H_n^*(A, W)$ is given by

$$H_n^*(A, W) \equiv \left(\frac{I(A = 1)}{g_n(1 \mid W)} - \frac{I(A = 0)}{g_n(0 \mid W)} \right).$$

Thus, for each subject with $A_i = 1$ in the observed data, we calculated the clever covariate as $H_n^*(1, W_i) = 1/g_n(1 \mid W_i)$. Similarly, for each subject with $A_i = 0$ in the observed data, we calculated the clever covariate as $H_n^*(0, W_i) = -1/g_n(0 \mid W_i)$. We combined these values to form a single column $H_n^*(A_i, W_i)$ in the data matrix. We also added two columns $H_n^*(1, W_i)$ and $H_n^*(0, W_i)$. The values for these columns were generated by setting $a = 0$ and $a = 1$. This is step 4 in Fig. 4.2.

Updating $\bar{Q}_n^0$. We then ran a logistic regression of our outcome Y on the clever covariate using as intercept the offset $\text{logit}\bar{Q}_n^0(A, W)$ to obtain the estimate ϵ_n, where ϵ_n is the resulting coefficient in front of the clever covariate $H_n^*(A, W)$. We next wanted to update the estimate $\bar{Q}_n^0$ into a new estimate $\bar{Q}_n^1$ of the true regression function $\bar{Q}_0$:

$$\text{logit}\, \bar{Q}_n^1(A, W) = \text{logit}\, \bar{Q}_n^0(A, W) + \epsilon_n H_n^*(A, W).$$

This parametric working model incorporated information from g_n, through $H_n^*(A, W)$, into an updated regression. One can now repeat this updating step by running a logisitic regression of outcome Y on the clever covariate $H_n^*(A, W)$ using as intercept the offset $\text{logit}\, \bar{Q}_n^1(A, W)$ to obtain the next update $\bar{Q}_n^2$. However, it follows that this time the coefficient in front of the clever covariate will be equal to zero, so that subsequent steps do not result in further updates. Convergence of the TMLE algorithm was achieved in one step. The TMLE of Q_0 was given by $Q_n^* = (\bar{Q}_n^1, Q_{W,n}^0)$. With ϵ_n, we were ready to update our prediction function at $a = 1$ and $a = 0$ according to the logistic regression working model. We calculated

$$\text{logit}\, \bar{Q}_n^1(1, W) = \text{logit}\bar{Q}_n^0(1, W) + \epsilon_n H_n^*(1, W),$$

for all subjects, and then

$$\text{logit}\, \bar{Q}_n^1(0, W) = \text{logit}\bar{Q}_n^0(0, W) + \epsilon_n H_n^*(0, W)$$

for all subjects and added a column for $\bar{Q}_n^1(1, W_i)$ and $\bar{Q}_n^1(0, W_i)$ to the data matrix. Updating $\bar{Q}_n^0$ is also illustrated in step 5 of Fig. 4.2.

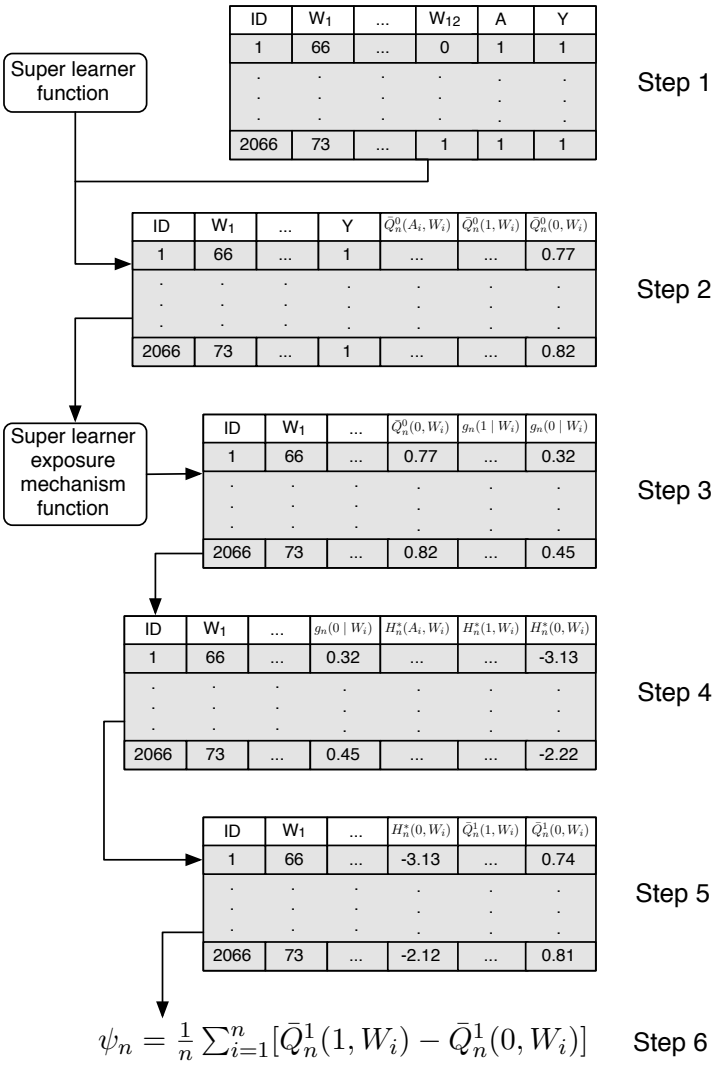

Fig. 4.2 Flow diagram for TMLE of the risk difference in the mortality study example

Targeted substitution estimator of the target parameter. We are at the last step! We computed the plug-in targeted maximum likelihood substitution estimator using the updated estimates $\bar{Q}_n^1(1, W)$ and $\bar{Q}_n^1(0, W)$ and the empirical distribution of W, as seen in step 6 of Fig. 4.2. Our formula from the first step becomes

$$\psi_{TMLE,n} = \Psi(Q_n^*) = \frac{1}{n}\sum_{i=1}^{n}\{\bar{Q}_n^1(1, W_i) - \bar{Q}_n^1(0, W_i)\}.$$

This mapping was accomplished by evaluating $\bar{Q}_n^1(1, W_i)$ and $\bar{Q}_n^1(0, W_i)$ for each observation i, and plugging these values into the above equation. Our estimate of the causal risk difference for the mortality study was $\psi_{TMLE,n} = -0.055$.

4.2.2 Inference

Standard errors. We then needed to calculate the influence curve for our estimator in order to obtain standard errors:

$$IC_n(O_i) = \left(\frac{I(A_i = 1)}{g_n(1 \mid W_i)} - \frac{I(A_i = 0)}{g_n(0 \mid W_i)}\right)(Y - \bar{Q}_n^1(A_i, W_i)) + \bar{Q}_n^1(1, W_i) - \bar{Q}_n^1(0, W_i) - \psi_{TMLE,n},$$

where I is an indicator function: it equals 1 when the logical statement it evaluates, e.g., $A_i = 1$, is true. Note that this influence curve is evaluated for each of the n observations O_i. The beauty of the influence curve of an estimator is that one can now proceed with statistical inference as if the estimator minus its estimand equals the empirical mean of the influence curve. Next, we calculated the sample mean of these estimated influence curve values: $\bar{IC}_n = \frac{1}{n}\sum_{i=1}^n IC_n(o_i)$, where we use o_i to stress that this mean is calculated with our observed realizations of the random variable O_i. For the TMLE we have $\bar{IC}_n = 0$. Using this mean, we calculated the sample variance of the estimated influence curve values:

$$S^2(IC_n) = \frac{1}{n}\sum_{i=1}^n \left(IC_n(o_i) - \bar{IC}_n\right)^2.$$

Lastly, we used our sample variance to estimate the standard error of our estimator:

$$\sigma_n = \sqrt{\frac{S^2(IC_n)}{n}}.$$

This estimate of the standard error in the mortality study was $\sigma_n = 0.012$.

Confidence intervals and p-values. With the standard errors, we can now calculate confidence intervals and p-values in the same manner you may have learned in other statistics texts. A 95% Wald-type confidence interval can be constructed as:

$$\psi_{TMLE,n} \pm z_{0.975}\frac{\sigma_n}{\sqrt{n}},$$

where z_α denotes the α-quantile of the standard normal density $N(0, 1)$. A p-value for $\psi_{TMLE,n}$ can be calculated as:

$$2\left[1 - \Phi\left(\left|\frac{\psi_{TMLE,n}}{\sigma_n/\sqrt{n}}\right|\right)\right],$$

where Φ denotes the standard normal cumulative distribution function. The p-value was < 0.001 and the confidence interval was $[-0.078, -0.033]$.

Interpretation

The interpretation of our estimate $\psi_{TMLE,n} = -0.055$, under causal assumptions, is that meeting or exceeding recommended levels of LTPA decreases 5-year mortality in an elderly population by 5.5%. This result was significant, with a p-value of < 0.001 and a confidence interval of $[-0.078, -0.033]$.

4.3 Practical Implications

The double robustness and semiparametric efficiency of the TMLE for estimating a target parameter of the true probability distribution of the data has important implications for both the analysis of RCTs and observational studies.

4.3.1 Randomized Controlled Trials

In 2010, a panel of the National Academy of Sciences made a recommendation to the FDA regarding the use of statistical methods for dealing with missing data in RCTs. The panel represented the split in the literature, namely, those supporting maximum-likelihood-based estimation, and specifically the use of multiple imputation (MI) methods, and the supporters of (augmented) inverse probability of censoring weighted (A-IPCW) estimators based on solving estimating equations. As a consequence, the committee's report ended up recommending both methods: a split decision.

Both camps at the table have been right in their criticism. The MI camp has been stating that the IPCW methods are too unstable and cannot be trusted in finite samples as demonstrated in various simulation studies, even though these methods can be made double robust. The A-IPCW camp has expressed that one cannot use methods that rely on parametric models that may cause severe bias in the resulting estimators of the treatment effect.

TMLE provides the solution to this problem of having to choose between two methods that have complementary properties: TMLE is a maximum-likelihood-based method and thus inherits all the attractive properties of maximum-likelihood-based substitution estimators, while it is still double robust and asymptotically efficient. TMLE has all the good properties of both the MI and the A-IPCW estimators, but it does not have the bad properties such as reliance on misspecified parametric models of the maximum-likelihood-based estimation the instability of the IPCW estimators due to not being substitution estimator. The FDA has also repeatedly ex-

pressed a desire for methods that can be communicated to medical researchers. As with maximum-likelihood-based estimation, the TMLE is easier to communicate: it is hard to communicate estimators that are defined as a solution of an estimating equation instead of a maximizer of a well-defined criterion.

TMLE can also be completely aligned with the highly populated maximum-likelihood-based estimation camp: TMLE can use maximum-likelihood-based estimation as the initial estimator, but it will carry out the additional targeting step. Of course, we recommend using the super learner (i.e., machine learning) as the initial estimator, but in an RCT in which one assumes that missingness is noninformative, the use of the parametric maximum likelihood estimation as initial estimator will not obstruct unbiased estimation of the causal effect of interest.

Consider an RCT in which we observe on each unit $(W, A, \Delta, \Delta Y)$, where Δ is an indicator of the clinical outcome being observed. Suppose we wish to estimate the additive causal effect $E_0Y_1 - E_0Y_0$, which is identified by the estimand $E_0[\bar{Q}_0(0, W) - \bar{Q}_0(1, W)]$, where $\bar{Q}_0(A, W) = E_0(Y \mid A, W, \Delta = 1)$ under causal assumptions, including that no unmeasured predictors of Y predict the missingness indicator. The TMLE of this additive causal effect only involves a minor modification of the TMLE presented above, and is derived in Appendix A. That is, the clever covariate is modified by multiplying it by $1/P_0(\Delta = 1 \mid A, W)$, and all outcome regressions are based on the complete observations only.

In an RCT the treatment assignment process, $g_0(1 \mid W) = P_0(A = 1 \mid W)$, is known (e.g., 0.5), and it is often assumed that missingness of outcomes is noninformative, also called missing completely at random. When this assumption holds, the g_n, comprising both the treatment assignment and the censoring or missingness mechanism, is always correctly estimated. Specifically, one can consistently estimate the missingness mechanism $P_0(\Delta = 1 \mid A, W)$ with the empirical proportions for the different treatment groups, thus ignoring the value of W. The TMLE will provide valid type I error control and confidence intervals for the causal effect of the investigated treatment, even if the initial regression estimator $\bar{Q}_n^0$ is completely misspecified.

The use of TMLE also often results in efficiency and bias gains with respect to the unadjusted or other ad hoc estimators commonly employed in the analysis of RCT data. For example, consider the additive causal effect example discussed in this chapter. The unadjusted estimator is restricted to considering only complete cases, ignoring observations where the outcome is missing, and ignoring any covariate information. In this particular example, the efficiency and bias gain is already apparent from the fact that the targeted maximum likelihood approach averages an estimate of an individual effect $\bar{Q}_0(1, W) - \bar{Q}_0(0, W)$ over all observations in the sample, including the observations that had a missing outcome.

TMLE can exploit information in measured baseline and time-dependent covariates, even when there is no missingness or right censoring. This allows for bias reduction due to empirical confounding, i.e., it will adjust for empirical imbalances in the treatment and control arm, and thereby improve finite sample precision (efficiency). To get an insight into the potential gains of TMLE relative to the current standard, we note that the relative efficiency of the TMLE relative to the unadjusted

estimator of the causal additive risk in a standard RCT with two arms and randomization probability equal to 0.5, and no missingness or censoring, is given by 1 minus the R squared of the regression of the clinical outcome Y on the baseline covariates W implied by the targeted maximum likelihood fit of the regression of Y on the binary treatment and baseline covariates. That is, if the baseline covariates are predictive, one will gain efficiency, and one can predict the amount of improvement from the actual regression fit.

Perhaps more importantly, the TMLE naturally adjusts for dropout (missingness) as well and can also be used to assess the effect of treatment under noncompliance, i.e., it is unbiased when standard methods are biased. Unlike an unadjusted estimator that ignores covariate information, TMLE does not rely on an assumption of noninformative missingness or dropout, but allows that missingness and dropout depend on the observed covariates, including time-dependent covariates.

In RCTs, including sequentially randomized controlled trials, one can still fully respect the likelihood of the data and obtain fully efficient and unbiased estimators, without taking the risk of bias due to statistical model misspecification (which has been the sole reason for the application of inefficient unadjusted estimators). On the contrary, the better one fits the true functions Q_0 and g_0, as can be evaluated with the cross-validated log-likelihood, the more bias reduction and efficiency gain will have been achieved.

Prespecification of the TMLE in the statistical analysis plan allows for appropriate adjustment with measured confounders while avoiding the possible introduction of bias should that decision be based on human intervention. Therefore, TMLEs can be used for both the efficacy as well as the safety analysis in Phase II, III, and IV clinical trials. In addition, just like for unadjusted estimators, permutation distributions can be used to obtain finite sample inference and more robust inference.

4.3.2 Observational Studies

At many levels of society one builds large electronic databases that keep track of large patient populations. One wishes to use these dynamic databases to assess safety signals of drugs, evaluate the effectiveness of different interventions, and so on. Comparative effectiveness research concerns the research involved to make such comparisons. These comparisons often involve observational studies, so that one cannot assume that the treatment was randomly assigned. In such studies, standard off-the-shelf methods are biased due to confounding as well as informative missingness, censoring, and possibly biased sampling.

In observational studies, the utilization of efficient and maximally unbiased estimators is thus extremely important. One cannot analyze the effect of high dose of a drug on heart attack in a postmarket safety analysis using logistic regression in a parametric statistical model or Cox proportional hazards models, and put much trust in a p-value. It is already a priori known that these statistical models are misspecified and that the effect estimate will be biased, so under the null hypothesis of no

treatment effect, the resulting test statistic will reject the null hypothesis incorrectly with probability tending to 1 as sample size increases. For example, if the high dose is preferentially assigned to sicker people, then the unadjusted estimator is biased high, a maximum likelihood estimator according to a misspecified parametric model will still be biased high by its inability to let the data speak and thereby adjust for the measured confounders.

As a consequence, the only alternative is to use semiparametric statistical models that acknowledge what is known and what is not known, and use robust and efficient substitution estimators. Given such infinite-dimensional semiparametric statistical models, we need to employ machine learning, and, in fact, as theory suggests, we should not be married to one particular machine learning algorithm but let the data speak by using super learning. That is, one cannot foresee what kind of algorithm should be used, but one should build a rich library of approaches, and use cross-validation to combine these estimators into an improved estimator that adapts the choice to the truth. In addition, and again as theory teaches us, we have to target the fit toward the parameter of interest, to remove bias for the target parameter, and to improve the statistical inference based on the central limit theorem. TMLE combined with super learning provides such a robust and semiparametric efficient substitution estimator, while we maintain the log-likelihood or other appropriate loss function as the principal criterion.

4.4 Summary

TMLE is a general algorithm where we start with an initial estimator of P_0, or a relevant parameter Q_0 of P_0. We then create a parametric statistical model with parameter ϵ through this given initial estimator whose score at $\epsilon = 0$ spans the efficient influence curve of the parameter of interest at the given initial estimator. It estimates ϵ with maximum likelihood estimation in this parametric statistical model and finally updates the new estimator as the corresponding fluctuation of the given initial estimator. The algorithm can be iterated until convergence, although in many common cases it converges in one step.

4.5 Road Map for Targeted Learning

We have now completed the road map for targeted learning depicted in Fig. 4.3. This chapter covered effect estimation using super learner and TMLE, as well as inference. In many cases, we may be interested in a ranked list of effect measures, often referred to as variable importance measures (VIMs). We provided an additional road map (Fig. 4.4) for research questions involving VIMs, which are common in medicine, genomics, and many other fields. We address questions of variable importance in Chaps. 22 and 23.

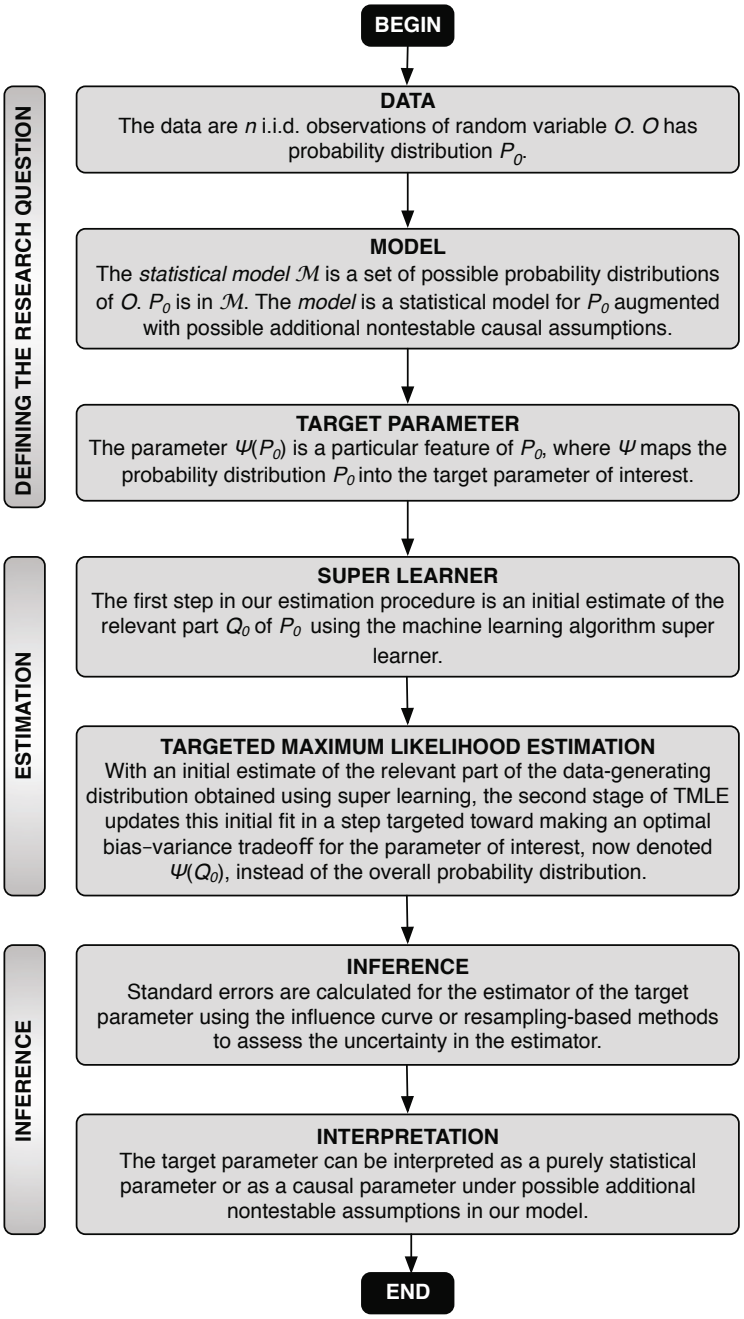

Fig. 4.3 Road map for targeted learning

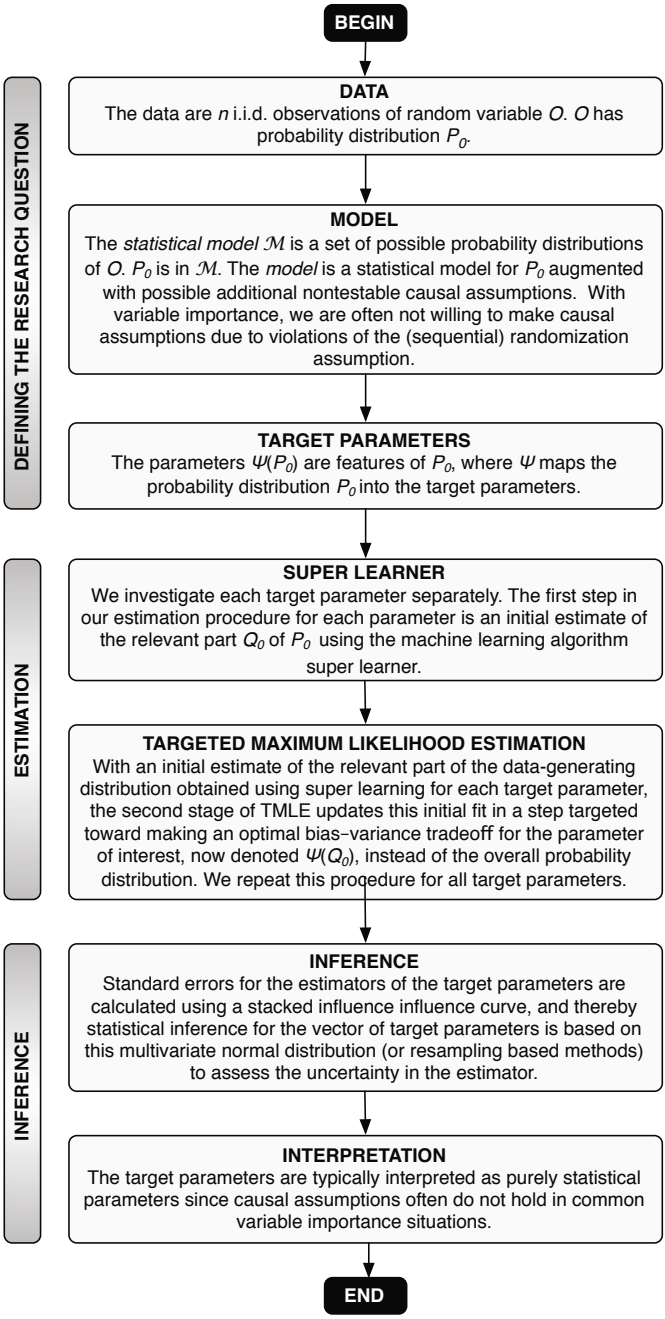

Fig. 4.4 Road map for targeted learning of variable importance measures

4.6 Notes and Further Reading

MLE has been referred to elsewhere as g-formula and g-computation. It is a maximum-likelihood-based substitution estimator of the g-formula parameter. The g-formula for identifying the distribution of counterfactuals from the observed data distribution, under the sequential randomization assumption, was originally published in Robins (1986). We also refer readers to an introductory implementation of a maximum-likelihood-based substitution estimator of the g-formula (Snowden et al. 2011; Rose et al. 2011).

Estimating equation methodology, including IPTW (Robins 1999b; Hernan et al. 2000) and A-IPTW (Robins et al. 2000b; Robins 2000; Robins and Rotnitzky 2001), is discussed in detail in van der Laan and Robins (2003). Detailed references and a bibliographic history on locally efficient A-IPTW estimators, double robustness, and estimating equation methodology can be found in Chap. 1 of that text. A key seminal paper in this literature is Robins and Rotnitzky (1992). A-IPTW was previously referred to as the double robust estimator in some publications. Didactic presentations of IPTW can be found in Robins et al. (2000a), Mortimer et al. (2005), and Cole and Hernan (2008).

For the original paper on TMLE we refer readers to van der Laan and Rubin (2006). Subsequent papers on TMLE in observational and experimental studies include Bembom and van der Laan (2007a), van der Laan (2008a), Rose and van der Laan (2008, 2009, 2011), Moore and van der Laan (2009a,b,c), Bembom et al. (2009), Polley and van der Laan (2009), Rosenblum et al. (2009), van der Laan and Gruber (2010), Gruber and van der Laan (2010a), Rosenblum and van der Laan (2010a), and Wang et al. (2010).

A detailed discussion of multiple hypothesis testing and inference for variable importance measures is presented in Dudoit and van der Laan (2008). We also refer readers to Chaps. 22 and 23. The mortality study analyzed in this chapter with TMLE is based on data discussed in Tager et al. (1998).

Previous work related to estimators in RCTs (and in general in observational studies with known probabilities of treatment) that are robust to model misspecification include, for example, Robins (1994), Robins et al. (1995), Scharfstein et al. (1999), van der Laan and Robins (2003), Leon et al. (2003), Tan (2006), Tsiatis (2006), Moore and van der Laan (2009b), Zhang et al. (2008), Rubin and van der Laan (2008), Freedman (2008a,b), and Rosenblum and van der Laan (2009a).

We refer readers to Bickel et al. (1997) for a text on semiparametric estimation and asymptotic theory. Tsiatis (2006) is a text applying semiparametric theory to missing data, including chapters on Hilbert spaces and influence curves. We also refer to Hampel et al. (1986) for a text on robust statistics, including presentation of influence curves. Van der Vaart (1998) provides a thorough introduction to asymptotic statistics, and van der Vaart and Wellner (1996) discuss stochastic convergence, empirical process theory, and weak convergence theory.

Chapter 5
Understanding TMLE

Sherri Rose, Mark J. van der Laan

This chapter focuses on understanding TMLE. We go into more detail than the previous chapter to demonstrate how this estimator is derived. Recall that TMLE is a two-step procedure where one first obtains an estimate of the data-generating distribution P_0 or the relevant portion Q_0 of P_0. The second stage updates this initial fit in a step targeted toward making an optimal bias–variance tradeoff for the parameter of interest $\Psi(Q_0)$, instead of the overall density P_0. The procedure is double robust and can incorporate data-adaptive-likelihood-based estimation procedures to estimate Q_0 and the treatment mechanism.

5.1 Conceptual Framework

We begin the discussion of TMLE at a conceptual level to give an overall picture of what the method achieves. In Fig. 5.1 we depict a flow chart for TMLE, and in this section, we walk the reader through the illustration and provide a conceptual foundation for TMLE. We start with our observed data and some (possibly) real valued function $\Psi()$, the target parameter mapping. These two objects are our inputs. We have an initial estimator of the probability distribution of the data (or something smaller than that – the relevant portion). This is P_n^0 and is estimated semiparametrically using super learning. This initial estimator is typically already somewhat informed about the target parameter of interest by, for example, only focusing on fitting the relevant part Q_0 of P_0. P_n^0 falls within the statistical model, which is the set of all possible probability distributions of the data. P_0, the true probability distribution, also falls within the statistical model, since it is assumed that the statistical model is selected to represent true knowledge. In many applications the statistical model is necessarily nonparametric. We update P_n^0 in a particular way, in a targeted way by incorporating the target parameter mapping Ψ, and now denote this targeted update as P_n^*. If we map P_n^* using our function $\Psi()$, we get our estimator $\Psi(P_n^*)$ and thereby a value on the real line. The updating step is tailored to result in values

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_5,

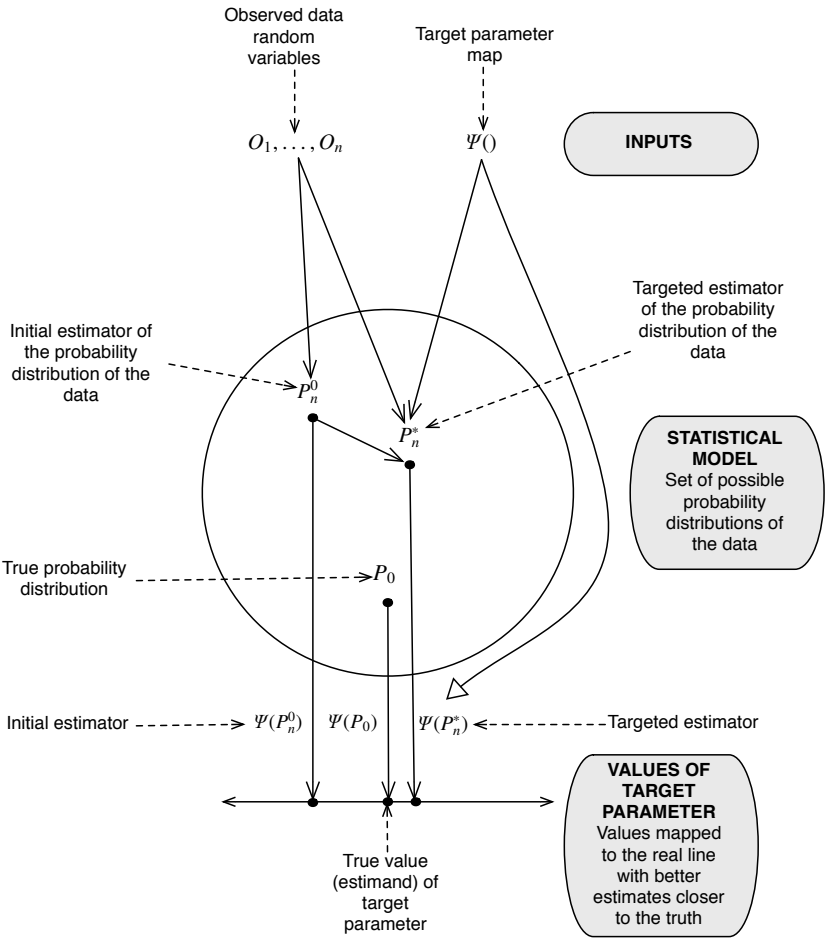

Fig. 5.1 TMLE flow chart.

$\Psi(P_n^*)$ that are closer to the truth than the value generated using the initial estimate P_n^0: specifically, $\Psi(P_n^*)$ is less biased than $\Psi(P_n^0)$.

TMLE provides a concrete methodology for mapping the initial estimator P_n^0 into a targeted estimator P_n^*, which is described below in terms of an arbitrary statistical model $\mathcal{M}$ and target parameter mapping $\Psi()$ defined on this statistical model. In order to make this more accessible to the reader, we then demonstrate this general template for TMLE with a nonparametric statistical model for a univariate random variable and a survival probability target parameter. Specifically, TMLE involves the following steps:

- Consider the target parameter $\Psi : \mathcal{M} \to \mathbb{R}$. Compute its pathwise derivative at P and its corresponding canonical gradient $D^*(P)$, which is also called the efficient

influence curve. This object $D^*(P)$, a function of O with mean zero under P, is now available for each possible probability distribution P.

- Define a loss function $L()$ so that $P \to E_0 L(P)$ is minimized at the true probability distribution P_0. One could select the log-likelihood loss function $L(P) = -\log P$. However, typically, this loss function is chosen so that it only depends on P through a relevant part $Q(P)$ and $Q \to L(Q)$ is minimized at $Q_0 = Q(P_0)$. This loss function could also be used to construct a super-learner-based initial estimator of Q_0.
- For a P in our model $\mathcal{M}$, define a parametric working model $\{P(\epsilon) : \epsilon\}$ with finite-dimensional parameter ϵ so that $P(\epsilon = 0) = P$, and a "score" $\frac{d}{d\epsilon} L(P(\epsilon))$ at $\epsilon = 0$ for which a linear combination of the components of this "score" equals the efficient influence curve $D^*(P)$ at P. Typically, we simply choose the parametric working model so that this score equals the efficient influence curve $D^*(P)$. If the loss function $L()$ only depends on P through a relevant part $Q = Q(P)$, then this translates into a parametric working model $\{Q(\epsilon) : \epsilon\}$ chosen so that a linear combination of the components of the "score" $\frac{d}{d\epsilon} L(Q(\epsilon))$ at $\epsilon = 0$ equals the efficient influence curve $D^*(P)$ at P.
- Given an initial estimator P_n^0 of P_0, we compute $\epsilon_n^0 = \arg\min_\epsilon \sum_{i=1}^n L(P_n^0(\epsilon))(O_i)$. This yields the first step TMLE $P_n^1 = P_n^0(\epsilon_n^0)$. This process is iterated: start with $k = 1$, compute $\epsilon_n^k = \arg\min_\epsilon \sum_{i=1}^n L(P_n^k(\epsilon))(O_i)$ and $P_n^{k+1} = P_n^k(\epsilon_n^k)$, increase k to $k + 1$, and repeat these updating steps until $\epsilon_n^k = 0$. The final update P_n^K at the final step K is denoted by P_n^* and is the TMLE of P_0. The same algorithm can be directly applied to Q_n^0 of $Q_0 = Q(P_0)$ for the case that the loss function only depends on P through $Q(P)$.
- The TMLE of ψ_0 is now the substitution estimator obtained by plugging P_n^* into the target parameter mapping: $\psi_n^* = \Psi(P_n^*)$. Similarly, if $\psi_0 = \Psi(Q_0)$ and the above loss function $L()$ is a loss function for Q_0, then we plug the TMLE Q_n^* into the target parameter mapping: $\psi_n^* = \Psi(Q_n^*)$.
- The TMLE P_n^* solves the efficient influence curve equation $0 = \sum_{i=1}^n D^*(P_n^*)(O_i)$, which provides a basis for establishing the asymptotic linearity and efficiency of the TMLE $\Psi(P_n^*)$.

For further presentation of TMLE at this general level we refer the interested reader to Appendix A.

Demonstration of TMLE template. In this section we demonstrate the TMLE template for estimation of survival probability. Suppose we observe n i.i.d. univariate random variables $O_1, \ldots, O_n$ with probability distribution P_0, where O_i represents a time to failure such as death. Suppose that we have no knowledge about this probability distribution, so that we select as statistical model the nonparametric model $\mathcal{M}$. Let $\Psi(P) = P(O > 5)$ be the target parameter that maps any probability distribution in its survival probability at 5 years, and let $\psi_0 = P_0(O > 5)$ be our target parameter of the true data-generating distribution.

The pathwise derivative $\Psi(P(\epsilon))$ at $\epsilon = 0$ for a parametric submodel (i.e., path) $\{P_S(\epsilon) = (1 + \epsilon S(P))P : \epsilon\}$ with univariate parameter ϵ is given by

$$\frac{d}{d\epsilon}\Psi(P_S(\epsilon))\Big|_{\epsilon=0} = E_P\{I(O > 5) - \Psi(P)\}S(P)(O).$$

Note that indeed, for any function S of O that has mean zero under P and is uniformly bounded, it follows that $P_S(\epsilon)$ is a probability distribution for a small enough choice of ϵ, so that the family of paths indexed by such functions S represents a valid family of submodels through P in the nonparametric model. By definition, it follows that the canonical gradient of this pathwise derivative at P (relative to this family of parametric submodels) is given by $D^*(P)(O) = I(O > 5) - \Psi(P)$. The canonical gradient is also called the efficient influence curve at P.

We could select the log-likelihood loss function $L(P) = -\log P(O)$ as loss function. A parametric working model through P is given by $P(\epsilon) = (1 + \epsilon D^*(P))P$, where ϵ is the univariate fluctuation parameter. Note that this parametric submodel includes P at $\epsilon = 0$ and has a score at $\epsilon = 0$ given by $D^*(P)$, as required for the TMLE algorithm. We are now ready to define the TMLE.

Let P_n^0 be an initial density estimator of the density P_0. Let

$$\epsilon_n^0 = \arg\max_{\epsilon} \sum_{i=1}^{n} \log P_n^0(\epsilon)(O_i),$$

and let $P_n^1 = P_n^0(\epsilon_n^0)$ be the corresponding first-step TMLE of P_0. It can be shown that the next iteration yields $\epsilon_n^1 = 0$, so that convergence of the iterative TMLE algorithm occurs in one step (van der Laan and Rubin 2006). The TMLE is thus given by $P_n^* = P_n^1$, and the TMLE of ψ_0 is given by the plug-in estimator $\psi_n^* = \Psi(P_n^*) = P_n^*(O > 5)$. Since P_n^* solves the efficient influence curve equation, it follows that $\psi_n^* = \frac{1}{n}\sum_{i=1}^{n} I(O_i > 5)$ is the empirical proportion of subjects that has a survival time larger than 5. This estimator is asymptotically linear with influence curve $D^*(P_0)$ since $\psi_n^* - \psi_0 = \frac{1}{n}\sum_{i=1}^{n} D^*(P_0)(O_i)$, which proves that the TMLE of ψ_0 is efficient for every choice of initial estimator: apparently, all bias of the initial estimator is removed by this TMLE update step.

Consider a kernel density estimator with an optimally selected bandwidth (e.g., based on likelihood-based cross-validation). Since this optimally selected bandwidth trades off bias and variance for the kernel density estimator as an estimate of the true density P_0, it will, under some smoothness conditions, select a bandwidth that converges to zero in sample size at a rate $n^{-1/5}$. The bias of such a kernel density estimator converges to zero at the rate $n^{-2/5}$. As a consequence, the substitution estimator of the survival function at t for this kernel density estimator has a bias that converges to zero at a slower rate than $1/\sqrt{n}$ in the sample size n. We can conclude that the substitution estimator of a survival function at 5 years based on this optimal kernel density estimator will have an asymptotic relative efficiency of zero (!) relative to the empirical survival function at 5 years. This simple example demonstrates that a regularized maximum likelihood estimator of P_0 is not targeted toward the target parameter of interest and, by the same token, that current Bayesian inference is not targeted toward the target parameter. However, if we apply the TMLE step to the kernel density estimator, then the resulting TMLE of the survival function is

unbiased and asymptotically efficient, and it even remains unbiased and asymptotically efficient if the kernel density estimator is replaced by an incorrect guess of the true density.

The point is: the best estimator of a density is not a good enough estimator of a particular feature of the density, but the TMLE step takes care of this.

5.2 Definition of TMLE in Context of the Mortality Example

This section presents the definition of TMLE in the context of our mortality example, thereby allowing the reader to derive the TMLE presented in the previous chapter. The reader may recognize the general recipe for TMLE as presented in Sect. 5.1 that can be applied in any semiparametric model with any target parameter. After having read this section, the reader might consider revisiting this general TMLE presentation. Our causal effect of interest is the causal risk difference, and the estimand is the corresponding statistical W-adjusted risk difference, which can be interpreted as the causal risk difference under causal assumptions. The data structure in the illustrative example is $O = (W, A, Y) \sim P_0$. TMLE follows the basic steps enumerated below, which we then illustrate in more detail.

TMLE for the Risk Difference

1. Estimate $\bar{Q}_0$ using super learner to generate our prediction function $\bar{Q}_n^0$. Let $Q_n^0 = (\bar{Q}_n^0, Q_{W,n})$ be the estimate of $Q_0 = (\bar{Q}_0, Q_{W,0})$, where $Q_{W,n}$ is the empirical probability distribution of $W_1, \ldots, W_n$.
2. Estimate the treatment mechanism using super learning. The estimate of g_0 is g_n.
3. Determine a parametric family of fluctuations $\{Q_n^0(\epsilon) : \epsilon\}$ of the initial estimator Q_n^0 with fluctuation parameter ϵ, and a loss function $L(Q)$ so that a linear combination of the components of the derivative of $L(Q_n^0(\epsilon))$ at $\epsilon = 0$ equals the efficient influence curve $D^*(Q_n^0, g_n)$ at any initial estimator $Q_n^0 = (\bar{Q}_n^0, Q_{W,n}^0)$ and g_n. Since the initial estimate $Q_{W,n}^0$ of the marginal distribution of W is the empirical distribution (i.e., nonparametric maximum likelihood estimator), the TMLE using a separate ϵ for fluctuating $Q_{W,n}^0$ and $\bar{Q}_n^0$ will only fluctuate $\bar{Q}_n^0$. The parametric family of fluctuations of $\bar{Q}_n^0$ is defined by parametric regression including a clever covariate chosen so that the above derivative condition holds with ϵ playing the role of the coefficient in front of the clever covariate. This "clever covariate" $H_n^*(A, W)$ depends on (Q_n^0, g_n) only through g_n, and in the TMLE procedure it needs to be evaluated for each observation (A_i, W_i), and at $(0, W_i)$, $(1, W_i)$.

4. Update the initial fit $\bar{Q}_n^0(A, W)$ from step 1. This is achieved by holding $\bar{Q}_n^0(A, W)$ fixed (i.e., as intercept) while estimating the coefficient ϵ for $H_n^*(A, W)$ in the parametric working model using maximum likelihood estimation. Let ϵ_n be this parametric maximum likelihood estimator. The updated regression is given by $\bar{Q}_n^1 = \bar{Q}_n^0(\epsilon_n)$. For the risk difference, no iteration is necessary, since the next iteration will not result in any change: that is, the next ϵ_n will be equal to zero. The TMLE of Q_0 is now $Q_n^* = (\bar{Q}_n^1, Q_{W,n}^0)$, where only the conditional mean estimator $\bar{Q}_n^0$ was updated.
5. Obtain the substitution estimator of the causal risk difference by application of the target parameter mapping to Q_n^*:

$$\psi_n = \Psi(Q_n^*) = \frac{1}{n} \sum_{i=1}^{n} \{\bar{Q}_n^1(1, W_i) - \bar{Q}_n^1(0, W_i)\}.$$

6. Calculate standard errors based on the influence curve of the TMLE ψ_n, and then calculate p-values and confidence intervals.

There are several concepts in this enumerated step-by-step list that may be somewhat opaque for the reader: the parametric working model coding the fluctuations of the initial estimator, the corresponding clever covariate, the efficient influence curve, and the influence curve. We expand upon the list, including these topics, below. For the nontechnical reader, we provide gray boxes so that you can read these to understand the essential topics relevant to each step. The white boxes outlined in black contain additional technical information for the more theoretical reader.

5.2.1 Estimating $\bar{Q}_0$

The first step in TMLE is obtaining an estimate $\bar{Q}_n^0$ for $\bar{Q}_0$. This initial fit is achieved using super learning, avoiding assuming a misspecified parametric statistical model.

5.2.2 Estimating g_0

The TMLE procedure uses the estimate of $\bar{Q}_0$ obtained above in conjunction with an estimate of g_0. We estimate g_0 with g_n, again using super learning.

5.2.3 Determining the Efficient Influence Curve $D^*(P)$

To obtain such a parametric working model to fluctuate the initial estimator Q_n^0 we need to know the efficient influence curve of the target parameter mapping at a particular P in the statistical model. This is a mathematical exercise that takes as input the definition of the statistical model $\mathcal{M}$ (i.e, the nonparametric model) and the target parameter mapping from this statistical model to the real line (i.e., $\Psi : \mathcal{M} \rightarrow \mathbb{R}$). We refer to Appendix A for required background material. It follows that the efficient influence curve at P_0 only depends on (Q_0, g_0) and is given by

$$D^*(Q_0, g_0)(W, A, Y) = \left(\frac{I(A = 1)}{g_0(1 \mid W)} - \frac{I(A = 0)}{g_0(0 \mid W)}\right)(Y - \bar{Q}_0(A, W)) + \bar{Q}_0(1, W) - \bar{Q}_0(0, W) - \Psi(Q_0).$$

More on the efficient influence curve. Calculation of the efficient influence curve, and of components of the efficient influence curve, requires calculations of projections of an element onto a subspace within a Hilbert space. These projections are defined in the Hilbert space $L_0^2(P)$ of functions of O that have mean zero under P endowed with an inner product $\langle S_1, S_2 \rangle_P = E_P S_1(O) S_2(O)$, being the covariance of two functions of O. Two elements in an Hilbert space are orthogonal if the inner product equals zero: so two functions of O are defined as orthogonal if their correlation or covariance equals zero. Recall that a projection of a function S onto a subspace of $L_0^2(P)$ is defined as follows: (1) the projection is an element of the subspace and (2) the difference of S minus the projection is orthogonal to the subspace. The subspaces on which one projects are so-called tangent spaces and subtangent spaces. The tangent space at P is defined as the closure of the linear span of all scores of submodels through P. The tangent space is a subspace of $L_0^2(P)$. The tangent space of a particular variation-independent parameter of P is defined as the closure of the linear span of all scores of submodels through P that only vary this particular factor. We can denote the tangent spaces by $T(P)$ and a projection of a function S onto a $T(P)$ by $\Pi(S \mid T(P))$.

5.2.4 Determining the Fluctuation Working Model

Now, can we slightly modify the initial estimator $\bar{Q}_n^0$ to reduce bias for the additive causal effect? Let $Q_{W,n}^0$ be the empirical probability distribution of $W_1, \ldots, W_n$. We refer to the combined conditional probability distribution of Y and the marginal probability distribution of W as Q_0. $Q_n^0 = (\bar{Q}_n^0, Q_{W,n}^0)$ denotes the initial estimator of this Q_0. We also remind the reader that the target parameter ψ_0 only depends on

P_0 through $\bar{Q}_0$ and $Q_{W,0}$. Since the empirical distribution $Q^0_{W,n}$ is already a nonparametric maximum likelihood estimator of the true marginal probability distribution of W, for the sake of bias reduction for the target parameter, we can focus on only updating $\bar{Q}^0_n$, as explained below.

We want to reduce the bias of our initial estimator, where the initial estimator is a random variable that has bias and variance. We only need to update $\bar{Q}^0_n$ since the empirical distribution $Q^0_{W,n}$ is a nonparametric maximum likelihood estimator (and can thus not generate bias for our target parameter).

Our parametric working model is denoted as $\{\bar{Q}^0_n(\epsilon) : \epsilon\}$, which is a small parametric statistical model, a one-dimensional submodel that goes through the initial estimate $\bar{Q}^0_n(A, W)$ at $\epsilon = 0$. If we use the log-likelihood loss function

$$L(\bar{Q})(O) = -\log \bar{Q}(A, W)^Y (1 - \bar{Q}(A, W))^{1-Y},$$

then the parametric working model for fluctuating the conditional probability distribution of Y, given (A, W), needs to have the property

$$\frac{d}{d\epsilon} \log \bar{Q}^0_n(\epsilon)(A, W)^Y (1 - \bar{Q}^0_n(A, W))^{1-Y}|_{\epsilon=0} = D^*_Y(Q^0_n, g_n)(W, A, Y), \tag{5.1}$$

where $D^*_Y(Q^0_n, g_n)$ is the appropriate component of the efficient influence curve $D^*(Q^0_n, g_n)$ of the target parameter mapping at (Q^0_n, g_n). Formally, the appropriate component D^*_Y is the component of the efficient influence curve that equals a score of a fluctuation of a conditional distribution of Y, given (A, W). These components of the efficient influence curve that correspond with scores of fluctuations that only vary certain parts of factors of the probability distribution can be computed with Hilbert space projections. We provide the required background and tools in Appendix A and various subsequent chapters.

More on fluctuating the initial estimator. If the target parameter ψ_0 depends on different variation-independent parts $(Q_{W,0}, \bar{Q}_0)$ of the probability distribution P_0, then one can decide to fluctuate the initial estimators $(Q^0_{W,n}, \bar{Q}_n)$ with separate submodels and separate loss functions $L(Q_W) = -\log Q_W$ and $L(\bar{Q})$, respectively. The submodels $\{Q^0_{W,n}(\epsilon) : \epsilon\}$, $\{\bar{Q}_n(\epsilon) : \epsilon\}$ and their corresponding loss functions $L(Q_W)$ and $L(\bar{Q})$ need to be chosen such that a linear combination of the components of the derivative $\frac{d}{d\epsilon} L(Q^0_n(\epsilon))\big|_{\epsilon=0}$ equals $D^*(Q^0_n, g_n)$ for the sum-loss function $L(Q) = L(Q_W) + L(\bar{Q})$. This corresponds with requiring that each of the two loss functions generates a "score" so that the sum of these two "scores" equals the efficient influence curve. If the initial estimator $Q^0_{W,n}$ is a nonparametric maximum likelihood estimator, the TMLE using a separate ϵ_1 and ϵ_2 for the two submodels will not update $Q^0_{W,n}$.

Following the protocol of TMLE, we also need to fluctuate the marginal distribution of W. For that purpose we select as loss function of $Q_{W,0}$ the log-likelihood loss function $-\log Q_W$. Then we would select a parametric working model coding fluctuations $Q^0_{W,n}(\epsilon)$ of $Q^0_{W,n}$ so that

$$\frac{d}{d\epsilon} \log Q^0_{W,n}(\epsilon)\Big|_{\epsilon=0} = D^*_W(Q^0_n, g_n),$$

where D^*_W is the component of the efficient influence curve that is a score of a fluctuation of the marginal distribution of W.

Tangent spaces. Since Q_W and $\bar{Q}$ represent parameters of different factors P_W and $P_{Y|A,W}$ in a factorization of $P = P_W P_{A|W} P_{Y|A,W}$, these components $D^*_W(P)$ and $D^*_Y(P)$ can be defined as the projection of the efficient influence curve $D^*(P)$ onto the tangent space of P_W at P and $P_{Y|A,W}$ at P, respectively. The tangent space T_W of P_W is given by all functions of W with mean zero. The tangent space T_Y of $P_{Y|A,W}$ is given by all functions of W, A, Y for which the conditional mean, given A, W, equals zero. The tangent space T_A of $P_{A|W}$ is given by all functions of A, W, with conditional mean zero, given W. These three tangent spaces are orthogonal, as a general consequence of the factorization of P into the three factors. The projection of a function S onto these three tangent spaces is given by $\Pi(S \mid T_W) = E_P(S(O) \mid W)$, $\Pi(S \mid T_Y)) = S(O) - E_P(S(O) \mid A, W)$, and $\Pi(S \mid T_A) = E_P(S(O) \mid A, W) - E_P(S \mid W)$, respectively. From these projection formulas and setting $S = D^*(P)$, the explicit forms of $D^*_W(P) = \Pi(D^*(P) \mid T_W)$ and $D^*_Y(P) = \Pi(D^*(P) \mid T_Y)$ can be calculated as provided below, and for each choice of P. It also follows that the projection of $D^*(P)$ onto the tangent space of $P_{A|W}$ equals zero: $\Pi(D^*(P) \mid T_A) = 0$. The latter formally explains that the TMLE does not require fluctuating the initial estimator of g_0. It follows that the efficient influence curve $D^*(P)$ at P can be decomposed as:

$$D^*(P) = D^*_Y(P) + D^*_W(P).$$

Our loss function for Q is now $L(Q) = L(\bar{Q}) + L(Q_W)$, and with this parametric working model coding fluctuations $Q^0_n(\epsilon) = (Q^0_{W,n}(\epsilon), \bar{Q}^0_n(\epsilon))$ of Q^0_n, we have that the derivative of $\epsilon \to L(Q^0_n(\epsilon))$ at $\epsilon = 0$ equals the efficient influence curve at (Q^0_n, g_n). If we use different ϵ for each component of Q^0_n, then the two derivatives span the efficient influence curve, since the efficient influence curve equals the sum of the two scores D^*_Y and D^*_W. Either way, the derivative condition is satisfied:

$$\langle \frac{d}{d\epsilon} L(Q^0_n(\epsilon))\Big|_{\epsilon=0} \rangle \supset D^*(Q^0_n, g_n), \tag{5.2}$$

where $D^*(Q_n^0, g_n) = D_Y^*(Q_n^0, g_n) + D_W^*(Q_n^0, g_n)$. Here we used the notation $\langle (h_1, \ldots, h_k) \rangle$ for the linear space consisting of all linear combinations of the functions $h_1, \ldots, h_k$. That is, the task of obtaining a loss function and parametric working model for fluctuating Q_n^0 so that the derivative condition holds has been completed.

Due to this property (5.2) of the parametric working model, the TMLE has the important feature that it solves the efficient influence curve equation $0 = \sum_i D^*(Q_n^*, g_n)(O_i)$ (also called the efficient score equation). Why is this true? Because at the next iteration of TMLE, the parametric maximum likelihood estimator $\epsilon_n = 0$, and a parametric maximum likelihood estimator solves its score equation, which exactly yields this efficient score equation. This is a strong feature of the procedure as it implies that TMLE is double robust and (locally) efficient under regularity conditions. In other words, TMLE is consistent and asymptotically linear if either Q_n or g_n is a consistent estimator, and if both estimators are asymptotically consistent, then TMLE is asymptotically efficient.

However, if one uses a separate ϵ_W and ϵ for the two parametric working models through $Q_{W,n}^0$ and $\bar{Q}_n^0$, respectively, then the maximum likelihood estimator of ϵ_W equals zero, showing that TMLE will only update $\bar{Q}_n^0$. Therefore, it was never necessary to update the part of Q_n^0 that was already nonparametrically estimated.

If the initial estimator of $Q_{W,0}$ is a nonparametric maximum likelihood estimator, then the TMLE does not update this part of the initial estimator Q_n^0.

Of course, we have not been explicit yet about how to construct this submodel $\bar{Q}_n^0(\epsilon)$ through $\bar{Q}_n^0$. For that purpose, we now note that $D_Y^*(Q_n^0, g_n)$ equals a function $H_n^*(A, W)$ times the residual $(Y - \bar{Q}_n^0(A, W))$, where

$$H_n^*(A, W) \equiv \left(\frac{I(A = 1)}{g_n(A = 1 \mid W)} - \frac{I(A = 0)}{g_n(A = 0 \mid W)} \right).$$

Here $I(A = 1)$ is an indicator variable that takes the value 1 when $A = 1$. One can see that for $A = 1$ the second term disappears, and for $A = 0$ the first term disappears.

It can be shown (and it is a classical result for parametric logistic main term regression in a parametric statistical model) that the score of a coefficient in front of a covariate in a logistic linear regression in a parametric statistical model for a conditional distribution of a binary Y equals the covariate times the residual. Therefore, we can select the following parametric working model for fluctuating the initial estimate of the conditional probability distribution of Y, given (A, W), or, equivalently, for the estimate of the probability of $Y = 1$, given (A, W):

$$\bar{Q}_n^0(\epsilon)(Y = 1 \mid A, W) = \frac{1}{1 + \exp\left(- \log \frac{\bar{Q}_n^0}{(1 - \bar{Q}_n^0)}(A, W) - \epsilon H_n^*(A, W) \right)}.$$

By this classical result, it follows that indeed the score of ϵ of this univariate logistic regression submodel at $\epsilon = 0$ equals $D_Y^*(Q_n^0, g_n)$. That is, we now have really fully succeeded in finding a parametric submodel through the initial estimator Q_n^0 that satisfies the required derivative condition. Since $H_n^*(A, W)$ now just plays the role of a covariate in a logistic regression, using an offset, this explains why we call the covariate $H_n^*(A, W)$ a clever covariate.

More on constructing the submodel. If one needs a submodel through an initial estimator of a conditional distribution of a binary variable Y, given a set of parent variables $Pa(Y)$, and it needs to have a particular score D_Y^*, then one can define this submodel as a univariate logistic regression model, using the initial estimator as offset, with univariate clever covariate defined as $H^*(Pa(Y)) = E(D_Y^* \mid Y = 1, Pa(Y)) - E(D_Y^* \mid Y = 0, Pa(Y))$. Application of this general result to the above setting yields the clever covariate $H^*(A, W)$ presented above.

If our goal was to target $P_0(Y_1 = 1)$ or $P_0(Y_0 = 1)$, then going through the same protocol for the TMLE shows that one would use as clever covariate

$$H_{0,n}^*(A, W) \equiv \left(\frac{I(A = 0)}{g_n(A = 0 \mid W)}\right) \text{ or } H_{1,n}^*(A, W) \equiv \left(\frac{I(A = 1)}{g_n(A = 1 \mid W)}\right).$$

By targeting these two parameters simultaneously, using a two-dimensional clever covariate with coefficients ϵ_1, ϵ_2, one automatically obtains a valid TMLE for parameters that are functions of these two marginal counterfactual probabilities, such as a causal relative risk and causal odds ratio.

By computing the TMLE that targets a multidimensional target parameter, one also obtains a valid TMLE for any (say) univariate summary measure of the multidimensional target parameter. By valid we mean that this TMLE will still satisfy the same asymptotic properties, such as efficiency and double robustness, as the TMLE that directly targets the particular summary measure. The TMLE that targets the univariate summary measure of the multidimensional parameter may have a better finite sample performance than the TMLE that targets the whole multidimensional target parameter, in particular, if the dimension of the multidimensional parameter is large.

5.2.5 Updating $\bar{Q}_n^0$

We first perform a logistic linear regression of Y on $H_n^*(A, W)$ where $\bar{Q}_n^0(A, W)$ is held fixed (i.e., used as an offset), and an additional intercept is suppressed in order to estimate the coefficient in front of $H_n^*(A, W)$, denoted ϵ. The TMLE procedure

is then able to incorporate information from g_n, through $H_n^*(A, W)$, into an updated regression. It does this by extracting ϵ_n, the maximum likelihood estimator of ϵ, from the fit described above, and updating the estimate $\bar{Q}_n^0$ according to the logistic regression working model. This updated regression is then given by $\bar{Q}_n^1$:

$$\text{logit } \bar{Q}_n^1(A, W) = \text{logit } \bar{Q}_n^0(A, W) + \epsilon_n H_n^*(A, W).$$

One iterates this updating process until the next $\epsilon_n = 0$ or has converged to zero, but, in this example, convergence is achieved in one step. The TMLE of Q_0 is now $Q_n^* = (Q_{W,n}^0, \bar{Q}_n^1)$. Note that this step is equivalent to $(\epsilon_{1n}, \epsilon_{2n}) = \arg\min_{\epsilon_1,\epsilon_2} \sum_i L(Q_n^0(\epsilon_1, \epsilon_2))(O_i)$, and setting $Q_n^1 = Q_n^0(\epsilon_{1n}, \epsilon_{2n})$, where, as noted above, $\epsilon_{1n} = 0$, so that only $\bar{Q}_n^0$ is updated.

Given a parametric working model $Q_n^0(\epsilon)$ with fluctuation parameter ϵ, and a loss function $L(Q)$ satisfying (5.2), the first-step TMLE is defined by determining the minimum ϵ_n^0 of $\sum_{i=1}^n L(Q_n^0(\epsilon))(O_i)$ and setting $Q_n^1 = Q_n^0(\epsilon_n^0)$. This updating process is iterated until convergence of $\epsilon_n^k = \arg\min_\epsilon \sum_{i=1}^n L(Q_n^k(\epsilon))$ to zero, and the final update Q_n^* is referred to as the TMLE of Q_0. In this case, the next $\epsilon_n^1 = 0$, so that convergence is achieved in one step and $Q_n^* = Q_n^1$.

5.2.6 Estimating the Target Parameter

The estimate $\bar{Q}_n^* = \bar{Q}_n^1$ obtained in the previous step is now plugged into our target parameter mapping, together with the empirical distribution of W, resulting in the targeted substitution estimator given by

$$\psi_n = \Psi(Q_n^*) = \frac{1}{n} \sum_{i=1}^n \{\bar{Q}_n^1(1, W_i) - \bar{Q}_n^1(0, W_i)\}.$$

This mapping is accomplished by evaluating $\bar{Q}_n^1(1, W_i)$ and $\bar{Q}_n^1(0, W_i)$ for each observation i and plugging these values into the above equation.

5.2.7 Calculating Standard Errors

The calculation of standard errors for TMLE can be based on the central limit theorem, relying on δ-method conditions. (See Appendix A for an advanced introduction to these topics.) Under such regularity conditions, the asymptotic behavior of the estimator, that is, its asymptotic normal limit distribution, is completely characterized

by the so-called influence curve of the estimator in question. In our example, we need to know the influence curve of the TMLE of its estimand.

Note that, in order to recognize that an estimator is a random variable, an estimator should be represented as a mapping from the data into the parameter space, where the data $O_1, \ldots, O_n$ can be represented by the empirical probability distribution function P_n. Therefore, let $\hat{\Psi}(P_n)$ be the TMLE described above. Since the TMLE is a substitution estimator, we have $\hat{\Psi}(P_n) = \Psi(P_n^*)$ for a targeted estimator P_n^* of P_0. An estimator $\hat{\Psi}(P_n)$ of ψ_0 is asymptotically linear with influence curve $IC(O)$ if it satisfies:

$$\sqrt{n}(\hat{\Psi}(P_n) - \psi_0) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} IC(O_i) + o_{P_0}(1).$$

Here the remainder term, denoted by $o_{P_0}(1)$, is a random variable that converges to zero in probability when the sample size converges to infinity. The influence curve $IC(O)$ is a random variable with mean zero under P_0.

More on estimators and the influence curve. An estimator $\hat{\Psi}(P_n)$ is a function $\hat{\Psi}$ of the empirical probability distribution function P_n. Specifically, one can express the estimator as a function $\hat{\Psi}$ of a large family of empirical means $1/n \sum_{i=1}^{n} f(O_i)$ of functions f of O varying over a class of functions $\mathcal{F}$. We say the estimator is a function of $P_n = (P_n f : f \in \mathcal{F})$, where we use the notation $P_n f \equiv 1/n \sum_{i=1}^{n} f(O_i)$. By proving that the estimator is a differentiable function $\hat{\Psi}$ of $P_n = (P_n f : f \in \mathcal{F})$ at $P_0 = (P_0 f : f \in \mathcal{F})$, and that a uniform central limit theorem applies to P_n based on empirical process theory, it follows that the estimator minus its estimand $\psi_0 = \hat{\Psi}(P_0)$ behaves in first order as an empirical mean of $IC(O_i)$: we write $\hat{\Psi}(P_n) - \psi_0 = (P_n - P_0)IC + o_P(1/\sqrt{n})$. This function $IC(O)$ is called the influence curve of the estimator, and it is uniquely determined by the derivative of $\hat{\Psi}$. Specifically, $IC(O) = \sum_{f \in \mathcal{F}} \frac{d}{dP_0 f} \hat{\Psi}((P_0 f : f)(f(O) - P_0 f)$, where, formally, the $\sum$ becomes an integral when $\mathcal{F}$ is not finite.

Asymptotic linearity is a desirable property as it indicates that the estimator behaves like an empirical mean, and, as a consequence, its bias converges to zero in sample size at a rate faster than $1/\sqrt{n}$, and, for n large enough, it is approximately normally distributed. The influence curve of an estimator evaluated as a function in O measures how robust the estimator is toward extreme values. The influence curve $IC(O)$ has mean zero under sampling from the true probability distribution P_0, and its (finite) variance is the asymptotic variance of the standardized estimator $\sqrt{n}(\hat{\Psi}(P_n) - \psi_0)$.

In other words, the variance of $\hat{\Psi}(P_n)$ is well approximated by the variance of the influence curve, divided by sample size n. If ψ_0 is multivariate, then the

covariance matrix of $\hat{\Psi}(P_n)$ is well approximated by the covariance matrix of the multivariate influence curve divided by sample size n. More importantly, the probability distribution of $\hat{\Psi}(P_n)$ is well approximated by a normal distribution with mean ψ_0 and the covariance matrix of the influence curve, divided by sample size.

An estimator is asymptotically efficient if its influence curve is equal to the efficient influence curve, $IC(O) = D^*(O)$. The influence curve of the TMLE indeed equals D^* if Q_n^* is a consistent estimator of Q_0, and g_n is a consistent estimator of g_0. A complete technical understanding of influence curve derivation is not necessary to implement the TMLE procedure. However, we provide Appendix A for a detailed methodology for deriving the influence curve of an estimator.

More on asymptotic linearity and efficiency. The TMLE is a consistent estimator of ψ_0 if either $\bar{Q}_n$ is consistent for $\bar{Q}_0$ or g_n is consistent for g_0. The TMLE is asymptotically linear under additional conditions. For a detailed theorem establishing asymptotic linearity and efficiency of the TMLE, we refer the reader to Chap. 27. In particular, if for some $\delta > 0$, $\delta < g_0(1 \mid W) < 1 - \delta$, and the product of the L^2-norm of $\bar{Q}_n - \bar{Q}_0$ and the L^2-norm of $g_n - g_0$ converges to zero at faster rate than $1/\sqrt{n}$, then the TMLE is asymptotically efficient. If g_n is a consistent estimator of g_0, then the influence curve of the TMLE $\hat{\Psi}(P_n)$ equals $IC = D^*(Q^*, g_0) - \Pi(D^*(Q^*, g_0) \mid T_g)$, the efficient influence curve at the possibly misspecified limit of Q_n^* minus its projection on the tangent space of the model for the treatment mechanism g_0. The projection term makes $D^*(Q^*, g_0)$ a conservative working influence curve, and the projection term equals zero if either $Q^* = Q_0$ or g_0 was known and $g_n = g_0$.

From these formal asymptotic linearity results for the TMLE it follows that if g_n is a consistent estimator of g_0, then the TMLE $\hat{\Psi}(P_n)$ is asymptotically linear with an influence curve that can be conservatively approximated by $D^*(Q^*, g_0)$, where Q^* denotes the possibly misspecified estimand of Q_n^*. If g_0 was known, as in a randomized controlled trial, and g_n was not estimated, then the influence curve of the TMLE equals $D^*(Q^*, g_0)$. If, on the other hand, g_n was estimated under a correctly specified model for g_0, then the influence curve of the TMLE has a smaller variance than the variance of $D^*(Q^*, g_0)$, except if $Q^* = Q_0$, in which case the influence curve of the TMLE equals the efficient influence curve $D^*(Q_0, g_0)$. As a consequence, we can use as a working estimated influence curve for the TMLE

$$IC_n(O) = \left(\frac{I(A = 1)}{g_n(1 \mid W)} - \frac{I(A = 0)}{g_n(0 \mid W)}\right)(Y - \bar{Q}_n^1(A, W)) + \bar{Q}_n^1(1, W) - \bar{Q}_n^1(0, W) - \psi_n.$$

Even if $\bar{Q}_n^1$ is inconsistent, but g_n is consistent, this influence curve can be used to obtain an asymptotically *conservative* estimator of the variance of the TMLE

$\hat{\Psi}(P_n)$. This is very convenient since the TMLE requires calculation of $D^*(Q_n^*, g_n)$, and apparently we can use the latter as influence curve to estimate the normal limit distribution of the TMLE.

If one assumes that g_n is a consistent maximum-likelihood-based estimator of g_0, then one can (asymptotically) conservatively estimate the variance of the TMLE with the sample variance of the estimated efficient influence curve $D^*(Q_n^*, g_n)$.

An estimate of the asymptotic variance of the standardized TMLE, $\sqrt{n}(\hat{\Psi}(P_n) - \psi_0)$, viewed as a random variable, using the estimate of the influence curve $IC_n(O)$ is thereby given by

$$\sigma_n^2 = \frac{1}{n} \sum_{i=1}^{n} IC_n^2(o_i).$$

5.3 Foundation and Philosophy of TMLE

TMLE in semiparametric statistical models for P_0 is the extension of maximum likelihood estimation in parametric statistical models. Three key ingredients are needed for this extension. Firstly, one needs to define the parameter of interest semiparametrically as a function of the data-generating distribution varying over the (large) semiparametric statistical model. Many practitioners are used to thinking of their parameter in terms of a regression coefficient, but that luxury is not available in semi- or nonparametric statistical models. Instead, one has to carefully think of what feature of the distribution of the data one wishes to target.

Secondly, one needs to estimate the true distribution P_0, or at least its relevant factor or portion as needed to evaluate the target parameter, and this estimate should respect the actual semiparametric statistical model. As a consequence, nonparametric maximum likelihood estimation is often ill defined or results in a complete overfit, and thereby results in estimators of the target parameter that are too variable. We discussed this issue in Chap. 3. The theoretical results obtained for the cross-validation selector (discrete super learner) inspired the general super learning methodology for estimation of probability distributions of the data, or factors of other high-dimensional parameters of the probability distributions of the data. In the sequel, a reference to a true probability distribution of the data is meant to refer to this relevant part of the true probability distribution of the data. This super learning methodology takes as input a collection of candidate estimators of the distribution of the data and then uses cross-validation to determine the best weighted combination of these estimators. It is assumed or arranged that the loss function is uniformly bounded so that oracle results for the cross-validation selector apply. The super learning methodology results now in an estimator of the distribution of the

data that will be used as an initial estimator in the TMLE procedure. The oracle results for this super learner teach us that the initial estimator is optimized with respect to a global loss function such as the log-likelihood loss function and is thereby not targeted toward the target parameter, $\Psi(P_0)$. That is, it will be too biased for $\Psi(P_0)$ due to a bias–variance tradeoff with respect to the more ambitious full P_0 (or relevant portion thereof) instead of having used a bias–variance tradeoff with respect to $\Psi(P_0)$. The targeted maximum likelihood step is tailored to remove bias due to the nontargeting of the initial estimator.

The targeted maximum likelihood step involves now updating this initial (super-learning-based) estimator P_n^0 of P_0 to tailor its fit to estimation of the target ψ_0, the value of the parameter $\Psi(P_0)$. This is carried out by determining a cleverly chosen parametric working model modeling fluctuations $P_n^0(\epsilon)$ of the initial estimator P_n^0 with a (say) univariate fluctuation parameter ϵ. The value $\epsilon = 0$ corresponds with no fluctuation so that $P_n^0(0) = P_n^0$. One now estimates ϵ with maximum likelihood estimation, treating the initial estimator as a fixed offset, and updates the initial estimator accordingly. If needed, this updating step is iterated to convergence, and the final update P_n^* is called the TMLE of P_0, while the resulting substitution estimator $\hat{\Psi}(P_n^*)$ of $\Psi(P_0)$ is the TMLE of ψ_0. This targeted maximum likelihood step thus uses a parametric maximum likelihood estimator, accordingly to a cleverly chosen parametric working model that includes the initial estimator, to obtain a bias reduction for the target $\Psi(P_0)$.

This is not just any parametric working model. That is, we wish to select a parametric working model such that the parametric maximum likelihood estimator is maximally effective in removing bias for the target parameter, at minimal increase in variance. So if ϵ_n is the parametric maximum likelihood estimator of ϵ, then we want the mean squared error of $\Psi(P_n^0(\epsilon_n)) - \psi_0$ to be as small as possible. We want this parametric working model to really listen to the information in the data that is relevant for the target parameter. In fact, we would like the parametric maximum likelihood estimator to be as responsive to the information in the data that is relevant for the target parameter as an estimator that is asymptotically efficient in the semiparametric model.

To get insight into what kind of choice of parametric working model may be as adaptive to such target-parameter-specific features in the data as a semiparametric efficient estimator, we make the following observations. Suppose one is interested in determining the parametric working model coding fluctuations $P_0(\epsilon)$ of P_0 so that the maximum likelihood estimator of $\psi_0 = \Psi(P_0(\epsilon = 0))$ according to this parametric working model is asymptotically equivalent to an efficient estimator in the large semiparametric model. Note that this parametric working model is not told that the true value of ϵ equals zero. It happens to be the case that from an asymptotic efficiency perspective this can be achieved as follows. Among all possible parametric working models that code fluctuations $P_0(\epsilon)$ of the true P_0 we chose the one for which the Cramer–Rao lower bound for the target parameter $\Psi(P_0(\epsilon))$ at $\epsilon = 0$ is equivalent to the semiparametric information bound for the target parameter at P_0. The Cramer–Rao lower bound for a parametric working model $P_0(\epsilon)$ is given by

$$\frac{\left\{\frac{d}{d\epsilon}\Psi(P_0(\epsilon))\Big|_{\epsilon=0}\right\}^2}{I(0)},$$

where $I(0)$ denotes the variance of the score of the parametric working model at $\epsilon = 0$. In parametric model theory $I(0)$ is called the information at parameter value 0. The semiparametric information bound for the target parameter at P_0 is defined as the supremum over all these possible Cramer–Rao lower bounds for the parametric working models. That is, the semiparametric information bound is defined as the Cramer–Rao lower bound for the hardest parametric working model. Thus, the parametric working model for which the parametric maximum likelihood estimator is as responsive to the data with respect to the target parameter as a semiparametric efficient estimator is actually given by this hardest parametric working model. Indeed, the TMLE selects this hardest parametric working model, but through P_n^0.

Note also that this hardest working parametric model can also be interpreted as the one that maximizes the change of the target parameter relative to a change $P_0(\epsilon) - P_0$ under small amounts of fluctuations. Thus this hardest working parametric model through an initial estimator P_n^0 will maximize the change of the target parameter relative to the initial value $\Psi(P_n^0)$ for small values of ϵ.

Beyond the practical appeal of this TMLE update that uses the parametric likelihood to fit the target parameter of interest, an important feature of the TMLE is that it solves the efficient influence curve equation, also called the efficient score equation, of the target parameter. We refer the reader to Sect. 5.2 and Appendix A for relevant material on the efficient influence curve. For now, it suffices to know that an estimator is semiparametric efficient if the estimator minus the true target parameter behaves as an empirical mean of $D^*(P_0)(O_i)$, $i = 1, \ldots, n$, showing the incredible importance of this transformation $D^*(P_0)$ of O, which somehow captures all the relevant information of O for the sake of learning the statistical parameter $\Psi(P_0)$. If $D^*(P)(O)$ is the efficient influence curve at P, a possible probability distribution for O in the statistical model, and P_n^* is the TMLE of P_0, then, $0 = \sum_{i=1}^n D^*(P_n^*)(O_i)$.

Just as a parametric maximum likelihood estimator solves a score equation by virtue of its maximizing the likelihood over the unknown parameters, a TMLE solves the target-parameter-specific score equation for the target parameter by virtue of maximizing the likelihood in a targeted direction. This can then be used to establish that the TMLE is asymptotically efficient if the initial estimator is consistent and remarkably robust in the sense that for many data structures and semiparametric statistical models, the TMLE of ψ_0 remains consistent even if the initial estimator is inconsistent. By using submodels that have a multivariate fluctuation parameter ϵ, the TMLE will solve the score equation implied by each component of ϵ. In this manner, one can obtain TMLEs that solve not only the efficient influence curve/efficient score equation for the target parameter, but also an equation that characterizes other interesting properties, such as being an imputation estimator (Gruber and van der Laan 2010a).

In particular, in semiparametric models used to define causal effect parameters, the TMLE is a double robust estimator. In such semiparametric models the probability distribution function P_0 can be factorized as $P_0(O) = Q_0(O)g_0(O)$, where g_0

is the treatment mechanism and Q_0 is the relevant factor that defines the g-formula for the counterfactual distributions. The TMLE $\Psi(Q_n^*)$ of $\psi_0 = \Psi(Q_0)$ is consistent if either Q_n^* or g_n is consistent. In our example, g_n is the estimator of the treatment mechanism $g_0(A \mid W) = P_0(A \mid W)$, and Q_n^* is the TMLE of Q_0.

5.4 Summary

TMLE of a parameter $\Psi(Q_0)$ distinguishes from nonparametric or regularized maximum likelihood estimation by fully utilizing the power of cross-validation (super learning) to fine-tune the bias–variance tradeoff with respect to the part Q_0 of the data-generating distribution, thereby increasing adaptivity to the true Q_0, and by targeting the fit to remove bias with respect to ψ_0. The loss-based super learner of Q_0 already outperforms with respect to bias and variance a regularized maximum likelihood estimator for the semiparametric statistical model with respect to estimation of Q_0 itself by its asymptotic equivalence to the oracle selector: one could include the regularized maximum likelihood estimator in the collection of algorithms for the super learner. Just due to using the loss-based super learner it already achieves higher rates of convergence for Q_0 itself, thereby improving both in bias and variance for Q_0 as well as $\Psi(Q_0)$. In addition, due to the targeting step, which again utilizes super learning for estimation of the required g_0 in the fluctuation function, it is less biased for ψ_0 than the initial loss-function-based super learner estimator, and, as a bonus, the statistical inference based on the central limit theorem is also heavily improved relative to just using a nontargeted regularized maximum likelihood estimator.

Overall it comes down to the following: the TMLE is a semiparametric efficient substitution estimator. This means it fully utilizes all the information in the data (super learning and asymptotic efficiency), in addition to fully using knowledge about global constraints implied by the statistical semiparametric statistical model $\mathcal{M}$ and the target parameter mapping (by being a substitution estimator), thereby making it robust under sparsity with respect to the target parameter. It fully incorporates the power of super learning for the benefit of getting closer to the truth in finite samples.

Chapter 6
Why TMLE?

Sherri Rose, Mark J. van der Laan

In the previous five chapters, we covered the targeted learning road map. This included presentation of the tools necessary to estimate causal effect parameters of a data-generating distribution. We illustrated these methods with a simple data structure: $O = (W, A, Y) \sim P_0$. Our target parameter for this example was $\Psi(P_0) = E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, which represents the causal risk difference under causal assumptions.

Throughout these chapters, the case for TMLE using super learning is compelling, but many of its properties have not been fully discussed, especially in comparison to other estimators. This chapter makes a comprehensive case for TMLE based on statistical properties and compares TMLE to maximum-likelihood-based substitution estimators of the g-formula (MLE) and estimating-equation-based methodology. We continue to refer to the simple data structure $O = (W, A, Y) \sim P_0$ and causal risk difference as the target parameter in some comparisons, but also discuss the performance of TMLE and other estimators globally, considering many target parameters and data structures.

As we introduced in Chaps. 4 and 5, TMLE has many attractive properties that make it preferable to other existing procedures for estimation of a target parameter of a data-generating distribution for arbitrary semiparametric statistical models. TMLE removes all the asymptotic residual bias of the initial estimator for the target parameter if it uses a consistent estimator of the treatment mechanism. If the initial estimator is already consistent for the target parameter, the minimal additional fitting of the data in the targeting step may potentially remove some finite sample bias and certainly preserve this consistency property of the initial estimator. As a consequence, TMLE is a so-called double robust estimator.

In addition, if the initial estimator and the estimator of the treatment mechanism are both consistent, then it is also asymptotically efficient according to semiparametric statistical model efficiency theory. That is, under this condition, other competing estimators will underperform in comparison for large enough sample sizes with respect to variance, assuming that the competitors are required to have a bias for the target parameter smaller than $1/\sqrt{n}$ across a neighborhood of distributions of the

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_6,

true P_0 that shrinks to P_0 at this same rate $1/\sqrt{n}$. It allows the incorporation of machine learning (i.e., super learning) methods for the estimation of both the relevant part of P_0 and the nuisance parameter g_0 required for the targeting step, so that we do not make assumptions about the probability distribution P_0 we do not believe. In this manner, every effort is made to achieve minimal bias and the asymptotic semiparametric efficiency bound for the variance. We further explain these issues in the pages that follow.

Portions of this chapter are technical, but a general understanding of the essential concepts can be gleaned from reading the introduction to each of the sections and the tables at the end of each section. For example, Sect. 6.1 explains that there are two general types of estimators and provides a list of various estimators that may be familiar to the reader. Similarly, Sects. 6.2–6.6 discuss properties of TMLE: it is a loss-based, well-defined, unbiased, efficient substitution estimator of target parameters of a data-generating distribution. The introductions explain these concepts and the closing tables summarize these properties among competing estimators and TMLE. Therefore, a strong math background is not required to understand the basic concepts, and some readers may find it useful to skim or skip certain subsections.

6.1 Landscape

In order to effectively establish the benefits of TMLE, we must enumerate competing estimators. For example, what are our competitors for the estimation of causal effect parameters, such as $E_0Y_1 - E_0Y_0$, as well as other target parameters? We group these estimators into two broad classes: MLE and estimating equation methodology. For each specific estimation problem, one can come up with a number of variations of an estimator in such a class. In Chaps. 7 and 21, among others, we provide a finite sample comparison of TMLE with a number of estimators, including estimators specifically tailored for this simple data structure. Recall that the conditional expectation of Y given (A, W) is denoted $E_0(Y \mid A, W) \equiv \bar{Q}_0(A, W)$. Additionally, we let $Q_n = (\bar{Q}_n, Q_{W,n})$ be the estimate of the conditional mean and the empirical distribution for the marginal distribution of W, representing the estimator of the true $Q_0 = (\bar{Q}_0, Q_W)$.

6.1.1 MLE

A maximum likelihood estimator for a parametric statistical model $\{p_\theta : \theta\}$ is defined as a maximizer over all densities in the parametric statistical model of the empirical mean of the log density:

$$\theta_n = \arg\max_\theta \sum_{i=1}^{n} \log p_\theta(O_i).$$

The $L(p)(O) = -\log p(O)$ is called a loss function at candidate density p for the true density p_0 since its expectation is minimized across all densities p by the true density $p = p_0$. This minimization property of the log-likelihood loss function is the principle behind maximum likelihood estimation providing the basis for establishing that maximum likelihood estimators for correctly specified statistical models approximate the true distribution P_0 for large sample size.

An estimator that is based on maximizing the log-likelihood over the whole statistical model or submodels of the statistical model or utilizes algorithms that involve maximization of the log-likelihood will be called a maximum-likelihood-based estimator. We use the abbreviation MLE to refer specifically to maximum-likelihood-based substitution estimators of the g-formula.

This chapter can be equally applied to the case where $L(p)(O)$ is replaced by any other loss function $L(Q)$ for a relevant part Q_0 of p_0, satisfying that $E_0 L(Q_0)(O) \leq E_0 L(Q)(O)$ for each possible Q. In that case, we might call this estimator a minimum-loss-based estimator. TMLE incorporates this case as well, in which it could be called targeted minimum-loss-based estimation (still abbreviated as TMLE). In this chapter we focus our comparison on the log-likelihood loss function and will thereby refer to MLE, including ML-based super learning.

The g-formula was previously discussed in Chaps. 1–4. Recall that uppercase letters represent random variables and lowercase letters are a specific value for that variable. $\Psi(P_0)$ for the causal risk difference can be written as the g-formula:

$$\Psi(P_0) = \sum_w \Big[\sum_y y P_0(Y = y \mid A = 1, W = w) - \sum_y y P_0(Y = y \mid A = 0, W = w) \Big] P_0(W = w), \tag{6.1}$$

where

$$P_0(Y = y \mid A = a, W = w) = \frac{P_0(W = w, A = a, Y = y)}{\sum_y P_0(W = w, A = a, Y = y)}$$

is the conditional probability distribution of $Y = y$, given $A = a$, $W = w$, and

$$P_0(W = w) = \sum_{y,a} P_0(W = w, A = a, Y = y).$$

Recall that our target parameter only depends on P_0 through the conditional mean $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$ and the marginal distribution Q_W of W; thus we can also write $\Psi(Q_0)$.

Maximum-likelihood-based substitution estimators of the g-formula are obtained by substitution of a maximum-likelihood-based estimator of Q_0 into the parameter mapping $\Psi(Q_0)$. The marginal distribution of W can be estimated with the nonparametric maximum likelihood estimator, which happens to be the empirical distribution that puts mass $1/n$ on each W_i, $i = 1, \ldots, n$. In other words, we estimate the expectation over W with the empirical mean over W_i, $i = 1, \ldots, n$. Maximum-

likelihood-based estimation of $\bar{Q}_0$ can range from the use of stratification to super learning. We introduced nonparametric estimation of $\bar{Q}_0$ in Chap. 3. Maximum-likelihood-based substitution estimators will be of the type

$$\psi_n = \Psi(Q_n) = \frac{1}{n}\sum_{i=1}^{n}\{\bar{Q}_n(1, W_i) - \bar{Q}_n(0, W_i)\}, \tag{6.2}$$

where this estimate is obtained by plugging in $Q_n = (\bar{Q}_n, Q_{W,n})$ into the mapping Ψ.

MLE using stratification. The simplest maximum likelihood estimator of $\bar{Q}_0$ stratifies by categories or possible values for (A, W). One then simply averages across the many categories (also called bins or treatment/covariate combinations). In most data sets, there will be a large number of categories with few or zero observations. One might refer to this as the curse of dimensionality, making the MLE for nonparametric statistical models typically ill defined, and an overfit to the data resulting in poor finite sample performance. One can refer to this estimator as the nonparametric MLE (NPMLE).

MLE after dimension reduction: propensity score methods. To deal with the curse of dimensionality, one might propose a dimension reduction W^r of W and apply the simple MLE to the reduced-data structure (W^r, A, Y). However, such a dimension reduction could easily result in a biased estimator of $\Psi(Q_0)$ by excluding confounders. One can show that a sufficient confounder is given by the propensity score $g_0(1 \mid W) = P_0(A = 1 \mid W)$, allowing one to reduce the dimension of W to only a single covariate, without inducing bias. A maximum likelihood estimator of $E_0(Y \mid A, W^r)$ can then be applied, where $W^r = g_0(1 \mid W)$, using stratification. For example, one creates five categories for the propensity score, thereby creating a total of ten categories for (A, W^r), and estimates $E_0(Y \mid A, W^r)$ with the empirical average of the outcomes within each category. Of course, this propensity score is typically unknown and will thus first need to be estimated from the data.

MLE using regression in a parametric working model. $\bar{Q}_0(A, W)$ is estimated using regression in a parametric working (statistical) model and plugged into the formula given in (6.2).

ML-based super learning. We estimate $\bar{Q}_0$ with the super learner, in which the collection of estimators may include stratified maximum likelihood estimators, maximum likelihood estimators based on dimension reductions implied by the propensity score, and maximum likelihood estimators based on parametric working models, beyond many other machine learning algorithms for estimation of $\bar{Q}_0$. Super learning requires a choice of loss function. If the loss function is a log-likelihood loss, $L(P_0)(O) = -\log p_0(O)$, then we would call this maximum-likelihood-based super learning. However, one might use a loss function for the relevant part $\bar{Q}_0$ that is not necessarily a log-likelihood loss, in which case we should call it minimum-loss-based super learning. For example, if Y is a continuous random variable with outcomes in $[0, 1]$, then one can select as loss function for $\bar{Q}_0$ the following function:

$$L(\bar{Q}_0)(O) = -Y \log \bar{Q}_0(A, W) + (1 - Y) \log(1 - \bar{Q}_0(A, W)),$$

which indeed satisfies that the expectation $E_0 L(\bar{Q})(O)$ is minimized by $\bar{Q} = \bar{Q}_0$. This loss function is an example of a loss function that is not a log-likelihood loss function. In this chapter we will not stress this additional important gain in generality of loss-based super learning relative to maximum-likelihood-based estimation, allowing us to proceed directly after the relevant parts of the distribution of P_0 required for evaluation of our target parameter $\Psi(P_0)$.

6.1.2 Estimating Equation Methods

Estimating-equation-based methodology for estimation of our target parameter $\Psi(P_0)$ includes inverse probability of treatment-weighted (IPTW) estimators and augmented IPTW (A-IPTW) estimators. These methods aim to solve an estimating equation in candidate ψ-values. An estimating function is a function of the data O and the parameter of interest. If $D(\psi)(O)$ is an estimating function, then we can define a corresponding estimating equation:

$$0 = \sum_{i=1}^{n} D(\psi)(O_i),$$

and solution ψ_n satisfying $\sum_{i=1}^{n} D(\psi_n)(O_i) = 0$. Most estimating functions for ψ will also depend on an unknown "nuisance" parameter of P_0. So we might define the estimating function as $D(\psi, \eta)$, where η is a candidate for the nuisance parameter. Given an estimator η_n of the required true nuisance parameter η_0 of P_0, we would define the estimating equation as

$$0 = \sum_{i=1}^{n} D(\psi, \eta_n)(O_i),$$

with solution ψ_n satisfying $\sum_{i=1}^{n} D(\psi_n, \eta_n)(O_i) = 0$. The theory of estimating functions teaches us that for each semiparametric statistical model and each target parameter, a class of estimating functions can be mathematically derived in terms of the gradients of the pathwise derivative of the target parameter, and the optimal estimating function that may yield an estimator with minimal asymptotic variance needs to be defined by the efficient influence curve (also called canonical gradient of the pathwise derivative) of the target parameter.

When the notation $D^*(\psi_0, \eta_0)$ is used for the estimating function $D(\psi_0, \eta_0)$, $D^*(\psi_0, \eta_0)$ is an estimating function implied by the efficient influence curve. An efficient influence curve is $D^*(P_0)(O)$, i.e., a function of O, but determined by P_0, and may be abbreviated $D^*(P_0)$ or $D^*(O)$. An optimal estimating function is one such that $D(\psi_0, \eta_0) = D^*(P_0)$.

For estimation of the causal risk difference, the following are two popular examples of estimating-equation-based methods, where the A-IPTW estimator is based on the estimating function implied by the efficient influence curve.

IPTW. One estimates our target parameter, the causal risk difference $\Psi(P_0)$, with

$$\psi_n = \frac{1}{n}\sum_{i=1}^{n}\{I(A_i = 1) - I(A_i = 0)\}\frac{Y_i}{g_n(A_i, W_i)}.$$

This estimator is a solution of an IPTW estimating equation that relies on an estimate of the treatment mechanism, playing the role of a nuisance parameter of the IPTW estimating function.

A-IPTW. One estimates $\Psi(P_0)$ with

$$\psi_n = \frac{1}{n}\sum_{i=1}^{n}\frac{\{I(A_i = 1) - I(A_i = 0)\}}{g_n(A_i, W_i)}(Y_i - \bar{Q}_n(A_i, W_i)) + \frac{1}{n}\sum_{i=1}^{n}\{\bar{Q}_n(1, W_i) - \bar{Q}_n(0, W_i)\}.$$

This estimator is a solution of the A-IPTW estimating equation that relies on an estimate of the treatment mechanism g_0 and the conditional mean $\bar{Q}_0$. Thus $(g_0, \bar{Q}_0)$ plays the role of the nuisance parameter of the A-IPTW estimating function. The A-IPTW estimating function evaluated at the true $(g_0, \bar{Q}_0)$ and true ψ_0 actually equals the efficient influence curve at the true data-generating distribution P_0, making it an optimal estimating function.

6.2 TMLE is Based on (Targeted) Loss-Based Learning

Suppose one is given a loss function $L()$ for a parameter $Q_0 = Q(P_0)$ while the estimand ψ_0 of interest is determined by Q_0. Thus, $Q_0 = \arg\min_Q E_0 L(Q)(O)$, where the minimum is taken over all possible parameter values of Q. One can proceed by defining a collection of candidate estimators $\hat{Q}_k$ that map the data P_n into an estimate of Q_0, where such estimators can be based on aiming to minimize the expected loss $Q \to E_0 L(Q)(O)$. This family of estimators can be used as a library of the loss-based super learner, which will use cross-validation to determine the best weighted combination of all these candidate estimators. The resulting super learner estimate Q_n can now be mapped into the estimate $\Psi(Q_n)$ of the estimand ψ_0.

Such estimators have the following properties. Firstly, these estimators are generally well defined by being based on minimizing empirical risk and cross-validated risk with respect to the loss function $L()$ over the statistical model. Secondly, by definition, these substitution estimators fully respect the global constraints implied by the statistical model and the target parameter mapping Ψ. Thirdly, such estimators can incorporate the state of the art in machine learning. Fourthly, the loss function

$L(Q)$ can be selected to result in good estimators of the estimand ψ_0: in particular, the TMLE chooses a loss function and a cleverly chosen parametric working model to construct a targeted loss function whose empirical risk represents the fit of the TMLE. Finally, such estimators can be constrained to also solve a particular estimating equation that might be considered to yield advantageous statistical properties of the substitution estimator of $\Psi(Q_n)$: The TMLE enforces such a constraint by iteratively minimizing the empirical risk over the parametric working model through the current initial estimate.

6.2.1 Competitors

MLE is a loss-based learning methodology based on the log-likelihood loss function $L(P_0) = -\log P_0$. This explains many of the popular properties of maximum-likelihood-based estimation. Since the log-likelihood loss function measures the performance of a candidate probability distribution as a whole, it does not represent a targeted loss function when the parameter of interest is a small feature of P_0. The lack of targeting of the MLE is particularly apparent when the data structure O is high dimensional and the statistical model is large.

An estimating equation method (e.g., A-IPTW) is not a loss-based learning method. It takes as input not a particular loss function but an estimating function, and the estimator is defined as a solution of the corresponding estimating equation. The estimating function is derived from local derivatives of the target parameter mapping and thereby ignores the global constraints implied by the statistical model and by the target parameter mapping. These global constraints are important to put a natural brake on estimators, so that it is no surprise that estimating equation methods are often notoriously unstable under sparsity.

6.2.2 TMLE

TMLE (targeted minimum-loss-based estimation) is a targeted-loss-based learning methodology. It is targeted by its choice of loss function $L()$ and by the targeted minimization over cleverly chosen parametric working models through an initial estimate. The TMLE is driven by the *global* choices of the loss function and parametric working model, and not defined by its consequence that it solves the efficient influence curve estimating equation, as implied by the *local* derivative condition. For example, consider the data structure $O = (W, A, Y)$, with Y continuous and bounded between 0 and 1.

Suppose that the statistical model is nonparametric and that the estimand is the additive treatment effect $E_{W,0}[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$, as in our mortality example. To define a TMLE we could select the squared error loss function $L(\bar{Q})(O) = (Y - \bar{Q}_0(A, W))^2$ for the conditional mean $\bar{Q}_0$, and the linear parametric

working model $\bar{Q}(\epsilon) = \bar{Q} + \epsilon H^*$. Alternatively, we could define a TMLE implied by the "quasi"-log-likelihood loss function $-Y \log \bar{Q}_0(A, W) - (1-Y)\log(1-\bar{Q}_0(A, W))$, and the logistic linear parametric working model $\text{logit}\bar{Q}(\epsilon) = \text{logit}\bar{Q} + \epsilon H^*$.

Both TMLEs solve the efficient influence curve estimating equation, but they have very different properties regarding utilization of the global constraints of the statistical model. The TMLE with the squared error loss does not respect that it is known that $P_0(0 < Y < 1) = 1$, and, as a consequence, the TMLE $\bar{Q}_n^*$ can easily predict far outside [0, 1], making it an unstable estimator under sparsity. In fact, this TMLE violates the very principle of TMLE in that TMLE should use a parametric *submodel* through the initial estimator, and the linear fluctuations of an initial estimator $\bar{Q}_n^0$ do *not* respect that $0 < \bar{Q}_0 < 1$, and are thus not a *submodel* of the statistical model. On the other hand, the other *valid* TMLE uses a logistic fluctuation of the initial estimator that fully respects this constraint, and is therefore a sensible substitution estimator fully respecting the global constraints of the statistical model. We refer to Chap. 7 for a full presentation of the latter TMLE for continuous and bounded Y.

Table 6.1 Summary of loss-based estimators for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Loss-based estimator of $\Psi(P_0)$	×	×	×	×	×		

6.3 TMLE Is Well Defined

An estimator that is well defined is desirable. Well-defined estimators have one solution in the space of possible solutions. It is easy to see why a well-defined estimator would be preferable to one that is not well defined. We seek the best estimate of $\Psi(P_0)$, and if our estimator gives multiple or no solutions, that presents a problem.

6.3.1 Competitors

MLEs aim to maximize a log-likelihood over candidate parameter values. Thus, MLE is often well defined, since, even if there are local maxima, the empirical log-likelihood or cross-validated log-likelihood can be used to select among such local maxima. Estimating equation methods are not well defined in general since the only criterion is that it solves the equation. A maximum likelihood estimator in a para-

metric statistical model often cannot be uniquely defined as a solution of the score equation since each local maximum will solve the score equation. The estimating equation methods are well defined for our target parameter with the simple data structure $O = (W, A, Y)$ and nonparametric statistical model for P_0, as is obvious from the definition of the IPTW and A-IPTW estimators given above. This is due to the fact that the estimating functions happen to be linear in ψ, allowing for a simple closed-form solution to their corresponding estimating equations.

When defining an estimator as a solution of the optimal efficient score/influence curve estimating equation, one may easily end up having to solve nonlinear equations that can have multiple solutions. The estimating equation itself provides no information about how to select among these candidate estimates of the target parameter. Also, one cannot use the likelihood since these estimators cannot be represented as $\hat{\Psi}(P_n)$ for some candidate P_n, i.e., these solutions ψ_n of the estimating equation are not substitution estimators (Sect. 6.6). This goes back to the basic fact that estimating functions (such as those defined by the efficient score/efficient influence curve) might not asymptotically identify the target parameter, and, even if they did, the corresponding estimating equation might not uniquely identify an estimator for a given finite sample.

In addition, for many estimation problems, the efficient influence curve $D^*(P_0)$ of the target parameter cannot be represented as an estimating function $D^*(\psi_0, \eta_0)$, so that the estimating equation methodology is not directly applicable. This means that the estimating equation methodology can only be applied if the efficient influence curve allows a representation as an estimating function. This is not a natural requirement, since the efficient influence curve $D^*(P_0)$ is defined as a gradient of the pathwise derivative of the target parameter along paths through P_0, and thereby only defines it as a function of P_0. There is no natural reason why the dependence of $D^*(P_0)$ on P_0 can be expressed in a dependence on two variation-independent parameters (ψ_0, η_0). Indeed, in some of our chapters we encounter target parameters where the efficient influence curve does not allow a representation as an estimating function.

6.3.2 TMLE

Unlike estimating function methodology (e.g., A-IPTW), TMLE does not aim to solve an estimating equation but instead uses the log-likelihood as a criterion. The super learner, representing the initial estimator in the TMLE, uses the (cross-validated) log-likelihood, or other loss function, to select among many candidate estimators. Even in the unlikely event that more than one global maximum exists, both would provide valid estimators so that a simple choice could make the super learner well defined. The targeting step involves computing a maximum likelihood estimator in a parametric working model of the same dimension as the target parameter, fluctuating the initial estimator, and is therefore as well defined as a parametric maximum likelihood estimator; again, the log-likelihood can be used to select

among different local maxima. See Table 6.2 for a summary of well-defined estimators of $\Psi(P_0)$.

Table 6.2 Summary of well-defined estimators for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Well-defined estimator of $\Psi(P_0)$	×	×	×	×	×		

6.4 TMLE Is Unbiased

An estimator is asymptotically unbiased if it is unbiased as the sample size approaches infinity. Bias is defined as follows: $\text{bias}(\psi_n) = E_0(\psi_n) - \psi_0$, where $E_0(\psi_n)$ denotes the expectation of the estimator ψ_n viewed as a function of the n i.i.d. copies $O_1, \ldots, O_n$ drawn from P_0. An estimator is unbiased if $\text{bias}(\psi_n) = 0$. It is rare that an estimator is exactly unbiased. If we restricted ourselves to using only unbiased estimators, then in most estimation problems we would have no estimators available. Therefore, one wants to focus on estimators where bias is negligible for the purpose of obtaining confidence intervals for ψ_0 and tests of null hypotheses about ψ_0. This can be achieved by requiring that the bias converge to zero when sample size n increases, at a rate smaller than $1/\sqrt{n}$, such as $1/n$. Indeed, most correctly specified parametric maximum likelihood estimators have a bias of the order $1/n$.

Why do we care about bias? In the real world, biased estimators can lead to false positives in multimillion-dollar studies. That is, the true causal risk difference might be equal to zero, but if the estimator is biased, then a test that ignores this bias will interpret the bias as a deviation from the null hypothesis. This deviation from the null hypothesis would be declared statistically significant if sample size was large enough. In addition, bias against the null hypothesis (for example, one wishes to test for a positive treatment effect, but the effect estimate is biased low) results in less power to reject the null hypothesis. Overall, bias causes incorrect statistical inference.

One might wonder why one would not aim to estimate the bias of an estimator. The problem is that estimation of bias is typically an impossible goal, inducing more error than the bias: often the best one can do is to diagnose the presence of unusual bias, and that is indeed a task that should be incorporated in a data analysis (Chap. 10). Again, our goal is the best estimator of the true effect, and an asymptotically biased estimator is an estimator that cannot even learn the truth. We also want the bias to be asymptotically negligible so that statistical assessment of uncertainty based on an estimator of the variance of the estimator is reasonably valid.

6.4.1 Competitors

MLEs using stratification, super learning, or parametric regression are asymptotically unbiased if $\bar{Q}_0$ is consistently estimated. In order for propensity score methods that fit a nonparametric regression on treatment A and the propensity score to be asymptotically unbiased, the estimator of g_0 must be consistent. MLEs using stratification can easily suffer from large finite sample bias in sparse data. In other words, using a nonparametric MLE with a limited data set provides no recipe for an unbiased estimator of the target parameter $\Psi(P_0)$. IPTW is asymptotically unbiased for $\Psi(P_0)$ if the estimator of g_0 is consistent, and A-IPTW is asymptotically unbiased for $\Psi(P_0)$ if either $\bar{Q}_0$ or g_0 is consistently estimated. The asymptotic bias of the A-IPTW is characterized by the same expression provided in the next paragraph for TMLE. The finite sample bias is very much a function of how g_0 is estimated, in particular with respect to what covariates are included in the treatment mechanism and how well it approximates the true distribution. Since g_n is by necessity estimated based on the log-likelihood for the treatment mechanism, its fit is not affected by data on Y. As a consequence, covariates that have no effect on Y but a strong effect on A will be included, only harming the bias reduction effort.

6.4.2 TMLE

Using super learning within TMLE makes our estimator of the outcome regression $\bar{Q}_0$ and estimator of the treatment mechanism g_0 maximally asymptotically unbiased. In our flexible nonparametric statistical model, we can show that the asymptotic bias in our procedure involves a product of the bias of $\bar{Q}_n^*$ and g_n relative to the true $\bar{Q}_0$ and g_0, respectively. For example, with data structure $O = (W, A, Y)$ in an observational study (where g_0 is unknown), our asymptotic bias of the TMLE $\Psi(Q_n^*)$ given by

$$\text{bias}(\psi_n) = P_0\Big\{\frac{g_0(1 \mid W) - g(1 \mid W)}{g(1 \mid W)}(\bar{Q}_0 - \bar{Q}^*)(1, W) - \frac{g_0(0 \mid W) - g(0 \mid W)}{g(0 \mid W)}(\bar{Q}_0 - \bar{Q}^*)(0, W)\Big\},$$

where $\bar{Q}^*$ and g denote the limits of $\bar{Q}_n^*$ and g_n. This teaches us that the asymptotic bias behaves as a second-order difference involving the product of approximation errors for g_0 and $\bar{Q}_0$. The empirical counterpart of this term plays the role of second-order term for the TMLE approximation of the true ψ_0, and thereby also drives the finite sample bias. For reliable confidence intervals one wants $\sqrt{n}$ times the empirical counterpart of this bias term to converge to zero in probability as sample size converges to infinity. If one wants to make this second-order term and the resulting bias as small as possible, then theory teaches us that we should use super learning

for both $\bar{Q}_0$ and g_0. As we point out in the next subsection, to minimize the variance of the first-order mean zero linear approximation of the TMLE approximation of the true ψ_0, one needs to estimate $\bar{Q}_0$ consistently. In other words, the use of super learning is essential for both maximizing efficiency as well as minimizing bias. For a formal theorem formalizing these statements we refer the interested reader to Chap. 27.

From this bias term one concludes that if the estimator of g_0 is correct, our estimator will have no asymptotic bias. This is an important scenario: consider again $O = (W, A, Y)$, and suppose we know the treatment mechanism, such as an RCT. In this instance, TMLE is always unbiased. Additionally, finite sample bias can be removed in RCTs by estimating g_0. If $\bar{Q}_n^*$ is already close to $\bar{Q}_0$, then the targeting step will further reduce the bias if g_n is also consistent. Finally, since running an additional univariate regression of the clever covariate on the outcome using the initial estimator as offset is a robust operation (assuming the clever covariate is bounded), even if g_n is misspecified, the targeting step will not cause harm to the bias.

In fact, one can show that if one replaces $g_0(A \mid W)$ by a true (sufficient) conditional distribution g_0^s of A, given a subset W^s of all covariates W, and W^s is chosen such that $Q^* - Q_0$ only depends on W through W^s, then the TMLE using this g_0 is also an unbiased estimator of the estimand ψ_0. Here Q^* represents the possibly misspecified estimand of the TMLE $\bar{Q}_n^*$. That is, the TMLE already achieves its full bias reduction by only incorporating the covariates in the treatment mechanism that explain the residual bias of $\bar{Q}_n^*$ with respect to $\bar{Q}_0$. We say that the TMLE is collaborative double robust to stress that consistency of the TMLE of ψ_0 is already achieved if g_n appropriately adapts to the residual bias of $\bar{Q}_n^*$: the TMLE is collaborative double robust, which is a stronger type of robustness with respect to misspecification of the nuisance parameters $\bar{Q}_0$ and g_0 than double robustness. In particular, an estimator g_n of g_0 used by the TMLE does not need to include covariates that are not predictive of Y, and are thus not confounders, even if the true treatment mechanism used these covariates. Apparently, the selection of covariates to be included in the estimator of the treatment mechanism should not be based on how well it fits g_0, but on the gain in fit of $\bar{Q}_0$ obtained by fitting the parametric working model (that uses this estimate of g_0) through the initial estimator $\bar{Q}_n^0$, relative to the fit of the initial estimator.

That is, TMLE naturally allows for the fine-tuning of the choice of g_n based on the fit of the corresponding TMLE of $\bar{Q}_0$, and can thereby data-adaptively select covariates into the treatment mechanism that actually matter and yield effective bias reduction in the TMLE step. For example, consider two possible estimators g_n^1 and g_n^2. These two choices combined with the initial estimator $\bar{Q}_n^0$ yield two different TMLEs, $\bar{Q}_{n1}^*$ and $\bar{Q}_{n2}^*$. These results suggest that one should select the estimator of g_0 for which the TMLE has the best fit of $\bar{Q}_0$. Note that this is equivalent to selecting covariates for the treatment mechanism based on how well the resulting estimate of the treatment mechanism improves the predictiveness of the corresponding clever covariate in predicting the outcome Y beyond the initial regression. This insight that the choice of g_n should be based on an evaluation of the resulting TMLE of $\bar{Q}_0$ is formalized by collaborative TMLE (C-TMLE), which is presented in Chaps. 19–21

and 23. See Table 6.3 for a summary of conditions for unbiased estimation among the estimators for a general $\Psi(P_0)$. Table 6.4 summarizes targeted estimation of the treatment mechanism for a general $\Psi(P_0)$.

Table 6.3 Summary of conditions for unbiased estimation for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Consistent estimation of $\bar{Q}_0$		×		×	×		
Consistent estimation of g_0			×			×	
Consistent estimation of $\bar{Q}_0$ or g_0	×						×
Problems in finite samples		×					

Table 6.4 Summary of targeted estimation of the treatment mechanism for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	(C-)TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Targeted estimation of treatment mechanism	×						

6.5 TMLE Is Efficient

Efficiency is another measure of the desirability of an estimator. Finite sample efficiency for an estimator ψ_n can be defined as

$$\text{efficiency}(\psi_n) = \frac{\left(\frac{1}{I(\Psi(P_0))}\right)}{n\text{var}(\psi_n)},$$

where $I(\Psi(P_0))$ is the Fisher information, defined as 1 over the variance of the efficient influence curve. The variance of the efficient influence curve is also called the generalized Cramer–Rao lower bound for the variance of locally (approximately) unbiased estimators. Thus, efficiency(ψ_n) is the ratio of the minimum possible asymptotic variance for an approximately unbiased estimator over its actual finite sample variance. The asymptotic efficiency is defined as the limit of efficiency(ψ_n) for n converging to infinity. If the estimator of $\Psi(P_0)$ is unbiased and the asymptotic efficiency(ψ_n) = 1, the estimator is asymptotically efficient. Asymptotically efficient estimators achieve the Cramer–Rao bound (i.e., the variance of an unbiased estimator is, at a minimum, the inverse of the Fisher information) for large n. What we really care about, though, is performance in finite samples. So we would like to see that the finite sample efficiency efficiency(ψ_n) is close to 1. Minimally, we want an asymptotically efficient estimator, but we also want our estimator to perform well in realistic finite sample sizes.

Efficiency theory is concerned with an admission criterion: it is restricted to only those estimators that have negligible bias (i.e., small bias in finite samples) along small fluctuations of the true data-generating distribution, and among such estimators it defines a best estimator as the estimator that has the smallest asymptotic variance. This best estimator will be asymptotically linear with influence curve the efficient influence curve $D^*(O)$. An estimator $\hat{\Psi}(P_n)$ of ψ_0 is asymptotically linear with influence curve $IC(O)$ if it satisfies

$$\sqrt{n}(\psi_n - \psi_0) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} IC(O_i) + o_{P_0}(1).$$

Here the remainder term, denoted by $o_{P_0}(1)$, is a random variable that converges to zero in probability when the sample size converges to infinity. Asymptotic linearity is a desirable property as it indicates that the estimator behaves like an empirical mean, and, as a consequence, its bias converges to zero in sample size at a rate faster than $1/\sqrt{n}$, and, for n large enough, it is approximately normally distributed. The influence curve of an estimator evaluated as a function in O measures how robust the estimator is toward extreme values. The influence curve $IC(O)$ has a mean of zero under sampling from the true probability distribution P_0, and its (finite) variance is the asymptotic variance of the standardized estimator $\sqrt{n}(\psi_n - \psi_0)$. In other words, the variance of $\hat{\Psi}(P_n)$ is well approximated by the variance of the influence curve, divided by sample size n. An estimator is asymptotically efficient if and only if its influence curve is equal to the efficient influence curve, $IC(O) = D^*(O)$.

If we already agree that we want unbiased estimators, why do we care about efficiency? Given two unbiased estimators why should we choose the one that is also efficient? An unbiased estimator that has a large spread (i.e., huge confidence intervals) may be uninformative. A practical real-world result of this, aside from improved interpretation, is huge potential cost savings. If we can extract more information out of the our data with an efficient estimator, we can reduce the sample size required for an inefficient estimator. This savings may be nontrivial. For example, in

a large multicenter RCT with a projected budget of $100 million, reducing sample size by 30% results in close to $30 million saved.

6.5.1 Competitors

If the covariate W is discrete, MLE using stratification is efficient asymptotically, but falls apart in finite samples if the number of categories is large. Suppose we have 30 discrete covariates, each with 3 levels. This gives us 3^{30} different covariate combinations, over 200 trillion! It is clear it becomes hopeless to wish for efficiency in finite sample sizes.

If W also includes continuous components, and some form of smoothing is used in the maximum likelihood estimation of $\bar{Q}_0$, then the maximum likelihood estimator will have approximation errors of the form $\sum_w E(\bar{Q}_n(1,w) - \bar{Q}_0(1,w))P_0(W = w)$ (minus the same term with $A = 0$). That is, the bias of $\bar{Q}_n$ will translate directly into a bias for the substitution estimator, and this bias will typically not be $o_P(1/\sqrt{n})$. The bias will also be larger than it would have been using super learning. In these cases, the bias causes the MLE to not be asymptotically linear and thereby also not achieve asymptotic efficiency. As discussed above, TMLE reduces the bias into a second-order term, so that it can still be asymptotically linear and thus efficient when the MLE will not (e.g., if g_0 can be well estimated).

Estimating equation methodology using the optimal estimating function (implied by the efficient influence curve) is asymptotically efficient if both $\bar{Q}_n$ and g_n are estimated consistently and if these estimators approximate the truth fast enough so that the estimator of ψ_0 succeeds in being asymptotically linear. This would require using super learning to estimate the nuisance parameters of the optimal estimating function. Due to the fact that estimating-equation-based estimators are not substitution estimators (Sect. 6.6), these estimators ignore global constraints, which harms the finite sample efficiency, in particular in the context of sparsity.

6.5.2 TMLE

Like the optimal estimating equation based estimator (i.e., A-IPTW), TMLE is double robust and (locally) efficient under regularity conditions. In other words, if the second-order term discussed above is asymptotically negligible, then the TMLE is consistent and asymptotically linear if either $\bar{Q}_n$ or g_n is a consistent estimator, and if both estimators are asymptotically consistent, then the TMLE is asymptotically efficient. TMLE also has excellent finite sample performance because it is driven by a log-likelihood (or other loss function) criterion, and a substitution estimator respecting all global constraints. The finite sample efficiency is further enhanced by the natural potential to fine-tune the estimator of the treatment mechanism through the predictiveness of the corresponding clever covariate, so that the treatment mech-

anism can be fitted in a way that is beneficial to its purpose in the targeting step. As previously noted, this is formalized by C-TMLE considered in later chapters. See Table 6.5 for a summary of efficiency among estimators for a general $\Psi(P_0)$.

Table 6.5 Summary of efficiency among estimators for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Efficient estimator of $\Psi(P_0)$	×	×					×
Problems in finite samples		×	×			×	×

6.6 TMLE Is a Substitution Estimator

Substitution estimators can be written as a mapping, taking an estimator of the relevant part of the data-generating distribution (e.g., P_n, P_n^*, Q_n, Q_n^*) and plugging it into the mapping $\Psi()$. The substitution estimator respects the statistical model space (i.e., the global constraints of the statistical model). Knowing and using information about the global constraints of the statistical model is helpful for precision (efficiency), particularly in the context of sparsity. For example, a substitution estimator of the risk difference ψ_0 respects knowledge that the mean outcome regression $\bar{Q}_0$ is bounded between $[0, 1]$, or that ψ_0 is a difference of two probabilities.

To understand why respecting global constraints in a statistical model is important in the context of sparsity (i.e., the data carry little information for target parameter), suppose one wishes to estimate the mean of an outcome Y based on observing n i.i.d. copies $Y_1, \ldots, Y_n$. Suppose it is also known that $E_0 Y$ is larger than 0 and smaller than 0.1. This knowledge is not needed if the sample size is large enough such that the standard error of the estimator is much smaller than 0.1, but for small sample sizes, it cannot be ignored.

6.6.1 Competitors

MLEs using stratification, super learning, propensity scores, and parametric regression are substitution estimators. An estimator of ψ_0 that is obtained as a solution of an estimating equation is often *not* a substitution estimator, i.e., it cannot be written as $\Psi(P_n)$ for a specified estimator P_n of P_0 in the statistical model. Indeed, IPTW

and A-IPTW are not substitution estimators. To be specific, suppose one wishes to estimate the treatment-specific mean $E_0Y_1 = E_0[E_0(Y \mid A = 1, W)]$ based on n i.i.d. copies of (W, A, Y), Y being binary. In this case, the A-IPTW estimator ψ_n, which solves the efficient influence curve estimating equation, can fall outside the range $[0, 1]$, due to inverse probability of treatments being close to zero. This proves that it is not a substitution estimator, which results in a loss of finite sample efficiency.

6.6.2 TMLE

The TMLE of ψ_0 is obtained by substitution of an estimator P_n^* into the mapping $\Psi()$. For the risk difference, this mapping is given in (6.1). As a consequence, it respects the knowledge of the statistical model. TMLE for the treatment-specific mean, discussed above, would result in E_0Y_1 between $[0, 1]$. See Table 6.6 for a summary of substitution estimators for a general $\Psi(P_0)$.

Table 6.6 Substitution estimators for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Substitution estimator of $\Psi(P_0)$	×	×	×	×	×		

6.7 Summary

The TMLE procedure produces a well-defined, unbiased, efficient substitution estimator of target parameters of a data-generating distribution. Competing estimators, falling into the broad classes of MLE and estimating equation methodology, do not have all of these properties and will underperform in many scenarios in comparison to TMLE. See Table 6.7 for a summary of statistical properties among estimators for a general $\Psi(P_0)$.

6.8 Notes and Further Reading

We refer readers to the references listed in Chap. 4. Appendix A covers further theoretical development of TMLE. A key reference for propensity score methods is Rosenbaum and Rubin (1983), and we also refer readers to Chap. 21.

Table 6.7 Summary of statistical properties among estimators for a general $\Psi(P_0)$

	MLE					**Estimating equations**	
	TMLE	Stratification	Propensity score	Parametric regression	ML-based super learning	IPTW	A-IPTW
Loss-based:							
Loss-based estimator of $\Psi(P_0)$	×	×	×	×	×		
Well-defined:							
Well-defined estimator of $\Psi(P_0)$	×	×	×	×	×		
Unbiased under:							
Consistent estimation of $\bar{Q}_0$		×		×	×		
Consistent estimation of g_0			×			×	
Consistent estimation of $\bar{Q}_0$ or g_0	×						×
Problems in finite samples		×					
Efficiency:							
Efficient estimator of $\Psi(P_0)$	×	×					×
Problems in finite samples		×	×			×	×
Substitution estimator:							
Substitution estimator of $\Psi(P_0)$	×	×	×	×	×		

Chapter 7
Bounded Continuous Outcomes

Susan Gruber, Mark J. van der Laan

This chapter presents a TMLE of the additive treatment effect on a bounded continuous outcome. A TMLE is based on a choice of loss function and a corresponding parametric submodel through an initial estimator, chosen so that the loss-function-specific score of this parametric submodel at zero fluctuation equals or spans the efficient influence curve of the target parameter. Two such TMLEs are considered: one based on the squared error loss function with a linear regression model, and one based on a quasi-log-likelihood loss function with a logistic regression submodel. The problem with the first TMLE is highlighted: the linear regression model is not a submodel and thus does not respect global constraints implied by the statistical model. It is theoretically and practically demonstrated that the TMLE with the logistic regression submodel is more robust than a TMLE based on least squares linear regression. Some parts of this chapter assume familiarity with the core concepts, as presented in Chap. 5. The less theoretically trained reader should aim to navigate through these parts and focus on the practical implementation and importance of the presented TMLE procedure. This chapter is adapted from Gruber and van der Laan (2010b).

7.1 Introduction

TMLE of a target parameter of the data-generating distribution, known to be an element of a semiparametric model, involves selecting a loss function (e.g., log-likelihood) and constructing a parametric *submodel* through an initial density estimator with parameter ϵ, so that the loss-function-specific "score" at $\epsilon = 0$ equals or spans the efficient influence curve (canonical gradient) at the initial estimator. This ϵ represents an amount of fluctuation of the initial density estimator. The latter "score" constraint can be satisfied by many loss functions and parametric submodels, since it represents only a local constraint of the submodels' behavior at zero fluctuation.

M.J. van der Laan and S. Rose, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer Series in Statistics, DOI 10.1007/978-1-4419-9782-1_7,

However, it is very important that the fluctuations encoded by the parametric model stay within the semiparametric model for the observed data distribution (otherwise it is not a submodel!), even if the target parameter can be defined on fluctuations that fall outside the assumed observed data model.

In particular, in the context of sparse data, by which we mean situations where the generalized Cramer–Rao lower bound is high, a violation of this property can significantly affect the performance of the estimator. We demonstrate this in the context of estimation of a causal effect of a binary treatment on a continuous outcome that is bounded. It results in a TMLE that inherently respects known bounds and consequently is more robust in sparse data situations than a TMLE using a naive parametric fluctuation working model that is actually not a *submodel* of the assumed statistical model.

Sparsity is defined as low information in a data set for the purpose of learning the target parameter. Formally, the Fisher information I is defined as sample size n divided by the variance of the efficient influence curve: $I = n/\text{var}(D^*(O))$, where $D^*(O)$ is the efficient influence curve of the target parameter at the true data-generating distribution. The reciprocal of the variance of the efficient influence curve can be viewed as the information one observation contains for the purpose of learning the target parameter. Since the variance of the efficient influence curve divided by n is the asymptotic variance of an asymptotically efficient estimator, one can also think of the information I as the reciprocal of the variance of an efficient estimator of the target parameter. Thus, sparsity with respect to a particular target parameter corresponds with small sample size relative to the variance of the efficient influence curve for that target parameter.

The following section begins with background on the application of TMLE methodology in the context of sparsity and its power relative to other semiparametric efficient estimators since it is a substitution estimator respecting global constraints of the semiparametric model. Even though an estimator can be asymptotically efficient without utilizing global constraints, the global constraints are instrumental in the context of sparsity with respect to the target parameter, motivating the need for semiparametric efficient *substitution* estimators, and for a careful choice of fluctuation function for the targeting step that fully respects these global constraints. A rigorous demonstration of the proposed TMLE of the causal effect of a binary treatment on a bounded continuous outcome follows, and the TMLE using a linear fluctuation function (i.e., that does not represent a parametric submodel) is compared with the proposed TMLE using a logistic fluctuation function. In Sect. 7.3, we carry out simulation studies that compare the two TMLEs of the causal effect, with and without sparsity in the data. Results for other commonly applied estimators discussed in Chap. 6 (MLE according to a parametric statistical model, IPTW, and A-IPTW) are also presented.

7.2 TMLE for Causal Estimation on a Continuous Outcome

We first review general TMLE so that we can clarify the important role of the choice of parametric working model, and thereby the fluctuation function, that defines the targeting update step of the initial estimator. Subsequently, in order to be specific, we define TMLE of the additive causal effect of a binary treatment on a bounded continuous outcome, which fully respects the known global bounds. Finally, we discuss its robustness in finite samples in the context of sparsity.

7.2.1 A Substitution Estimator Respecting the Statistical Model

A TMLE is a semiparametric efficient substitution estimator of a target parameter $\Psi(P_0)$ of a true distribution $P_0 \in \mathcal{M}$, known to be an element of a statistical model $\mathcal{M}$, based on sampling n i.i.d. $O_1, \ldots, O_n$ from P_0. Firstly, one notes that $\Psi(P_0) = \Psi(Q_0)$ only depends on P_0 through a relevant part $Q_0 = Q(P_0)$ of P_0. Secondly, one proposes a loss function $L(Q)$ such that

$$Q_0 = \arg\min_{Q \in \mathcal{Q}} E_0 L(Q)(O),$$

where $\mathcal{Q} = \{Q(P) : P \in \mathcal{M}\}$ is the set of possible values for Q_0. Thirdly, one uses minimum-loss-based learning, such as super learning, fully utilizing the power and optimality results for loss-based cross-validation to select among candidate estimators, to obtain an initial estimator Q_n^0 of Q_0. Fourthly, one proposes a parametric fluctuation $Q_{g_n,n}^0(\epsilon)$, possibly indexed by the estimator g_n of nuisance parameter $g_0 = g(P_0)$, such that

$$\frac{d}{d\epsilon} L(Q_{g_n,n}^0(\epsilon))(O)\Big|_{\epsilon=0} = D^*(Q_n^0, g_n)(O), \tag{7.1}$$

where $D^*(P) = D^*(Q(P), g(P))$ is the efficient influence curve of the pathwise derivative of the statistical target parameter mapping $\Psi : \mathcal{M} \to \mathbb{R}$ at $P \in \mathcal{M}$. If a multivariate ϵ is used, then the derivatives with respect to each of their components ϵ_j must span the efficient influence curve $D^*(Q_n^0, g_n)$. Fifthly, one computes the amount of fluctuation with minimum-loss-based estimation:

$$\epsilon_n = \arg\min_{\epsilon} \sum_{i=1}^{n} L(Q_{g_n,n}^0(\epsilon))(O_i).$$

This yields an update $Q_n^1 = Q_{g_n,n}^0(\epsilon_n)$. This updating of an initial estimator Q_n^0 into a next Q_n^1 is iterated until convergence, resulting in a final update Q_n^*. Since at the last step the amount of fluctuation $\epsilon_n \approxeq 0$, this final Q_n^* will solve the efficient influence curve estimating equation:

$$0 = \frac{1}{n}\sum_{i=1}^{n} D^*(Q_n^*, g_n)(O_i),$$

representing a fundamental ingredient for establishing the asymptotic efficiency of $\Psi(Q_n^*)$. Recall that an estimator is efficient if and only if it is asymptotically linear with an influence curve equal to the efficient influence curve $D^*(Q_0, g_0)$. Finally, the TMLE of ψ_0 is the substitution estimator $\Psi(Q_n^*)$.

Thus we see that TMLE involves constructing a parametric submodel $\{Q_n^0(\epsilon) : \epsilon\}$, and thereby its corresponding fluctuation function $\epsilon \rightarrow Q_n^0(\epsilon)$, through the initial estimator Q_n^0 with parameter ϵ, where the score of this parametric submodel at $\epsilon = 0$ equals the efficient influence curve at the initial estimator. The latter constraint can be satisfied by many parametric submodels, since it represents only a local constraint of its behavior at zero fluctuation. However, it is very important that the fluctuations stay within the statistical model for the observed data distribution, even if the target parameter Ψ can be defined on fluctuations of densities that fall outside the assumed observed data model. In particular, in the context of sparse data (i.e., data that will not allow for precise estimation of the target parameter), a violation of this property can significantly affect the performance of the estimator.

One important strength of the semiparametric efficient TMLE relative to the alternative semiparametric efficient estimating equation methodology is that it respects the global constraints of the observed data model. This is due to the fact that it is a substitution estimator $\Psi(Q_n^*)$ with Q_n^*, an estimator of a relevant part Q_0 of the true distribution of the data in the observed data model. The estimating equation methodology does not result in substitution estimators and consequently often ignores important global constraints of the observed data model, which comes at a price in the context of sparsity. Indeed, simulations have confirmed this gain of TMLE relative to the efficient estimating equation method in the context of sparsity (see Chap. 20 and also Stitelman and van der Laan 2010), which is demonstrated in this chapter. However, if TMLE violates the principle of being a substitution estimator by allowing Q_n^* to fall outside the assumed observed data model, this advantage is compromised. Therefore, it is crucial that TMLE use a fluctuation function that is guaranteed to map the fluctuated initial estimator into the statistical model.

7.2.2 Procedure

To demonstrate the important consideration of selecting a fluctuation function in the construction of TMLE that corresponds with a parametric *sub*model, we consider the problem of estimating the additive causal effect of a binary treatment A on a continuous outcome Y, based on observing n i.i.d. copies of $O = (W, A, Y) \sim P_0$, where W is the set of confounders. Consider the following SCM: $W = f_W(U_W)$, $A = f_A(W, U_A)$, $Y = f_Y(W, A, U_Y)$ with the functions f_W, f_A, and f_Y unspecified, representing a set of assumptions about how O is generated. We assume that U_A is independent of U_Y such that the randomization assumption $(A \perp Y_a \mid W)$ holds

with respect to the counterfactuals $Y_a = f_Y(W, a, U_Y)$ as defined by this SCM. In this SCM for the data-generating distribution of the observed data O, the additive causal effect $E_0(Y_1 - Y_0)$ can be identified from the observed data distribution through the statistical parameter of P_0:

$$\Psi(P_0) = E_0[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)].$$

Suppose that it is known that $Y \in [a, b]$ for some $a < b$. Alternatively, one might have truncated the original data to fall in such an interval and focus on the causal effect of treatment on this truncated outcome, motivated by the fact that estimating the conditional means of unbounded, or very heavy tailed, outcomes requires very large data sets. The SCM implies no assumptions about the statistical model $\mathcal{M}$ so that the statistical model is nonparametric. The target parameter mapping $\Psi : \mathcal{M} \to \mathbb{R}$ and the estimand $\psi_0 = \Psi(P_0)$ are now defined. The statistical estimation problem is to estimate ψ_0 based on observing n i.i.d. copies $O_1, \ldots, O_n$.

Let $Y^* = (Y - a)/(b - a)$ be the linearly transformed outcome within $[0, 1]$, and we define the statistical parameter

$$\Psi^*(P_0) = E_0[E_0(Y^* \mid A = 1, W) - E_0(Y^* \mid A = 0, W)],$$

which can be interpreted as the causal effect of treatment on the bounded outcome Y^* in the postulated SCM. We note the following relation between the causal effect on the original outcome Y and the causal effect on the transformed outcome Y^*:

$$\Psi(P_0) = (b - a)\Psi^*(P_0).$$

An estimate, normal limit distribution, and confidence interval for $\Psi^*(P_0)$ is now immediately mapped into an estimate, normal limit distribution, and confidence interval for $\Psi(P_0)$ by simple multiplication. Suppose $\sqrt{n}(\psi_n - \Psi^*(P_0)) \xrightarrow{d} N(0, \sigma^{2*})$, then $\sqrt{n}((b-a)\psi_n - \Psi(P_0)) \xrightarrow{d} N(0, \sigma^2)$, with $\sigma^2 = (b-a)^2\sigma^{2*}$. Upper and lower bounds on the confidence interval for $\Psi^*(P_0)$, given as (c^*_{lb}, c^*_{ub}), are multiplied by $(b - a)$ to obtain upper and lower bounds on $\Psi(P_0)$, $c_{lb} = (b-a)c^*_{lb}$, and $c_{ub} = (b-a)c^*_{ub}$. As a consequence, for notational convenience, without loss of generality, we can assume $a = 0$ and $b = 1$ so that $Y \in [0, 1]$.

To determine a loss function and corresponding fluctuation function, and thereby the definition of the TMLE, we need to know the efficient influence curve. The efficient influence curve of the statistical parameter $\Psi : \mathcal{M} \to \mathbb{R}$, defined on a nonparametric statistical model $\mathcal{M}$ for P_0 at the true distribution P_0, is given by

$$D^*(P_0) = \frac{2A - 1}{g_0(A \mid W)}(Y - \bar{Q}_0(A, W)) + \bar{Q}_0(1, W) - \bar{Q}_0(0, W) - \Psi(Q_0), \qquad (7.2)$$

where $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$ and $Q_0 = (Q_{W,0}, \bar{Q}_0)$ denotes both this conditional mean $\bar{Q}_0$ and the marginal distribution $Q_{W,0}$ of W. Note that indeed $\Psi(P_0)$ only depends on P_0 through the conditional mean $\bar{Q}_0$ and the marginal distribution of W. We will use the notation $\Psi(P_0)$ and $\Psi(Q_0)$ interchangeably. Note also that the

efficient influence curve only depends on P_0 through Q_0, g_0, so that we will also denote the efficient influence curve $D^*(P_0)$ with $D^*(Q_0, g_0)$. In order to stress that $D^*(P_0)$ can also be represented as an estimating function in ψ, we also now and then denote it by $D^*(Q_0, g_0, \psi_0)$.

We are ready to define a TMLE of $\Psi(Q_0)$, completely analogous to the TMLE presented in Chaps. 4 and 5 for a binary outcome. Let $\bar{Q}_n^0$ be an initial estimate of $\bar{Q}_0(A, W) = E_0(Y \mid A, W)$ with predicted values in $(0, 1)$. This could be a loss-based super learner based on the squared error loss function or the quasi-log-likelihood loss function presented below. In addition, we estimate $Q_{W,0}$ with the empirical distribution of $W_1, \ldots, W_n$. Let Q_n^0 denote the resulting initial estimate of Q_0. The targeting step will also require an estimate g_n of $g_0 = P_{A|W}$. As we will see, only the estimate $\bar{Q}_n^0$ of the conditional mean $\bar{Q}_0$ will be modified by the TMLE procedure defined below: this makes sense since the empirical distribution of W is already a nonparametric maximum likelihood estimator so that no bias gain with respect to the target parameter will be obtained by modifying it.

We use as fluctuation function for the empirical distribution $Q_{W,n}$, $Q_{W,n}(\epsilon_1) = (1 + \epsilon_1 D_2^*(Q_n^0))Q_{W,n}$, where $D_2^*(Q_n^0) = \bar{Q}_n^0(1, W) - \bar{Q}_n^0(0, W) - \Psi(Q_n^0)$ is the second component of the efficient influence curve $D^*(Q_n^0, g_n)$. We use the log-likelihood loss function, $-\log Q_W$, as loss function for the marginal distribution of W. It follows that

$$\frac{d}{d\epsilon} \log Q_{W,n}(\epsilon_1)\Big|_{\epsilon_1=0} = D_2^*(Q_n^0),$$

showing that this fluctuation function and log-likelihood loss function for the marginal distribution of W indeed generates the wished score at zero fluctuation.

We can represent the estimate $\bar{Q}_n^0$ as

$$\bar{Q}_n^0 = \frac{1}{1 + \exp(-f_n^0)},$$

with $f_n^0 = \log(\bar{Q}_n^0/(1 - \bar{Q}_n^0))$. Consider now the following fluctuation function:

$$\bar{Q}_n^0(\epsilon_2) = \frac{1}{1 + \exp(-\{f_n^0 + \epsilon_2 H_{g_n}^*\})},$$

which maps a fluctuation parameter value ϵ_2 into a modification $\bar{Q}_n^0(\epsilon_2)$ of the initial estimate. This fluctuation function is indexed by a function

$$H_{g_n}^*(A, W) = \frac{2A - 1}{g_n(A \mid W)}.$$

Equivalently, we can write this fluctuation function in terms of fluctuations of the logit of $\bar{Q}_n^0$: $\text{logit}\bar{Q}_n^0(\epsilon_2) = \text{logit}\bar{Q}_n^0 + \epsilon_2 H^*(g_n)$.

Consider now the following quasi-log-likelihood loss function for the conditional mean $\bar{Q}_0$:

$$-L(\bar{Q})(O) = Y \log \bar{Q}(A, W) + (1 - Y) \log(1 - \bar{Q}(A, W)).$$

Note that this is the log-likelihood of the conditional distribution of a binary outcome Y, but now extended to continuous outcomes in $[0, 1]$. It is thus known that this loss function is a valid loss function for the conditional distribution of a binary Y, but we need it to be a valid loss function for a conditional mean of a continuous $Y \in [0, 1]$. It is indeed a valid loss function for the conditional mean of a continuous outcome in $[0, 1]$, as has been previously noted. See Wedderburn (1974) and McCullagh (1983) for earlier uses of logistic regression for continuous outcomes in $[0, 1]$. We formally prove this result in Lemma 7.1 at the end of this chapter. The proposed fluctuation function $\bar{Q}_n^0(\epsilon_2)$ and the quasi-log-likelihood loss function satisfy

$$\frac{d}{d\epsilon_2} L(\bar{Q}_n^0(\epsilon_2))\Big|_{\epsilon_2=0} = H^*(A, W)(Y - \bar{Q}_n^0(A, W)),$$

giving us the desired first component $D_1^*(\bar{Q}_n^0, g_n)$ of the efficient influence curve $D^* = D_1^* + D_2^*$, where $D_2^*(Q_0) = \bar{Q}_0(1, W) - \bar{Q}_0(0, W) - \Psi(Q_0)$.

Our combined loss function is given by $L(Q) = -\log Q_W + L(\bar{Q})$, and, for $\epsilon = (\epsilon_1, \epsilon_2)$, our parametric fluctuation function for the combined Q is given by $Q(\epsilon) = (Q_W(\epsilon_1), \bar{Q}(\epsilon_2))$. With these choices of loss function $L(Q)$ for Q_0 and fluctuation function $Q(\epsilon)$ of Q, we indeed now have that

$$\frac{d}{d\epsilon_j} L(Q(\epsilon))\Big|_{\epsilon=0} = D_j^*(Q, g), \ j = 1, 2.$$

This shows that we succeeded in defining a loss function for $Q_0 = (Q_{W,0}, \bar{Q}_0)$ and fluctuation function such that the derivatives as defined in (7.1) span the efficient influence curve. The TMLE is now defined!

In this first targeting step, the maximum likelihood estimator of ϵ_1 equals zero, so that the update of $Q_{W,n}$ equals $Q_{W,n}$ itself. As a consequence of $\epsilon_{1,n}^0 = 0$ being the maximum likelihood estimator, the empirical mean of the component $D_2^*(Q_n^*) = \bar{Q}_n^*(1, W) - \bar{Q}_n^*(0, W) - \Psi(Q_n^*)$ of the efficient influence curve at the final TMLE equals zero: of course, this is trivially verified.

The maximum likelihood estimator of ϵ_2 for fluctuating $\bar{Q}_n^0$ is given by

$$\epsilon_{2n}^0 = \underset{\epsilon_2}{\operatorname{argmin}} \, P_n L(\bar{Q}_n^0(\epsilon_2)),$$

where we used the notation $P_n f = 1/n \sum_i f(O_i)$. This "maximum likelihood" estimator of ϵ_2 can be computed with generalized linear regression using the binomial link, i.e., the logistic regression maximum likelihood estimation procedure, simply ignoring that the outcome is not binary, which also corresponds with iterative reweighted least squares estimation using iteratively updated estimated weights of the form $1/(\bar{Q}_n(1 - \bar{Q}_n))$.

This provides us with the targeted update $Q_n^1 = Q_n^0(\epsilon_n^0)$, where the empirical distribution of W was not updated, but $\bar{Q}_n^0$ did get updated to $\bar{Q}_n^0(\epsilon_n^0)$. Iterating this procedure now defines the TMLE Q_n^*, but, as in the binary outcome case, we have that $\bar{Q}_n^2 = \bar{Q}_n^1(\epsilon_n^1) = \bar{Q}_n^1$ since the next maximum likelihood estimator $\epsilon_n^1 = 0$, and, of

course, the maximum likelihood estimator of ϵ_1 remains 0. Thus convergence occurs in one step, so that $Q_n^* = Q_n^1$. The TMLE of ψ_0 is thus given by $\Psi(Q_n^*) = \Psi(Q_n^1)$. As a consequence of the definition of the TMLE, we have that the TMLE Q_n^* solves the efficient influence curve estimating equation $P_n D^*(Q_n^*, g_n, \Psi(Q_n^*)) = 0$.

7.2.3 Robustness of TMLE in the Context of Sparsity

We note that, even if there is strong confounding causing some large values of $H_{g_n}^*$, the resulting TMLE $\bar{Q}_n^*$ remains bounded in $(0, 1)$, so that the TMLE $\Psi(Q_n^*)$, which just averages values of $\bar{Q}_n^*$, fully respects the global constraints of the observed data model. An inspection of the efficient influence curve (7.2), $D^*(P_0)$, reveals that there are two potential sources of sparsity. Small values for $g_0(A \mid W)$ and large outlying values of Y inflate the variance. Enforcing (e.g., known) bounds on Y and g_0 in the estimation procedure provides a means for controlling these sources of variance. We note that, even if there is strong confounding causing some large values of $h_{g_n^0}$, the resulting TMLE $\bar{Q}_n^*$ remains bounded in $(0, 1)$, so that the TMLE $\Psi(Q_n^*)$ fully respects the global constraints of the observed data model. On the other hand, the A-IPTW estimator obtained by solving the efficient influence curve estimating equation, $P_n D^*(Q_n^0, g_n, \psi) = 0$, in ψ yields the estimator

$$\psi_n = \frac{1}{n} \sum_{i=1}^{n} H_{g_n}^*(A_i, W_i)(Y_i - \bar{Q}_n^0(A_i, W_i)) + \bar{Q}_n^0(1, W) - \bar{Q}_n^0(0, W).$$

This estimator can easily fall outside $[0, 1]$ if for some observations $g_n(1 \mid W_i)$ is close to 1 or 0. This represents the price of not being a substitution estimator.

It is also important to contrast this TMLE with the TMLE using the linear fluctuation function. The latter TMLE would use the $L(\bar{Q}) = (Y - \bar{Q}(A, W))^2$ loss function, and fluctuation function $\bar{Q}_n^0(\epsilon) = \bar{Q}_n^0 + \epsilon H^*(g_n)$, so that (7.1) is still satisfied. The TMLE is defined as above, and again converges in one step. One estimates the fluctuation ϵ with univariate least squares linear regression, using $\bar{Q}_n^0$ as offset. In this case, large values of $H^*(g_n)$ will result in predicted values of $\bar{Q}_n^0(\epsilon_n)$ that are outside the bounds $[a, b]$. Therefore, this version of TMLE does not repect the global constraints of the model, i.e., the knowledge that $Y \in [a, b]$. In the next section, an analysis of a simulated data set provides a comparison of TMLE using the logistic fluctuation function and TMLE using this linear fluctuation.

7.3 Simulations

Two simulation studies illustrate the effects of employing a logistic vs. linear fluctuation function in the definition of the TMLE. These two studies evaluate practical performance with and without sparsity in the data, where a high degree of sparsity

corresponds to a target parameter that is borderline identifiable. As above, the parameter of interest is defined as the additive effect of a binary point treatment on the outcome, $\psi_0 = E_0[E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W)]$. We also implement three additional estimators: MLE, IPTW, and A-IPTW.

7.3.1 Estimators

In the simulation setting, Y is not bounded, so that we do not have an a priori a and b bound on Y. Instead of truncating Y and redefining the target parameter as the causal effect on the truncated Y, we still aim to estimate the causal effect on the original Y. Therefore, in the TMLE using a logistic fluctuation function we set $a = \min(Y)$, $b = \max(Y)$, and $Y^* = (Y - a)/(b - a)$. In this TMLE, the initial estimate $\bar{Q}_n^{0,Y^*}$ of $E_0(Y^*|A, W)$ needs to be represented as a logistic function of its logit transformation. Note that $\text{logit}(x)$ is not defined when $x = 0$ or 1. Therefore, in practice $\bar{Q}_n^{0,Y^*}$ needs to be bounded away from 0 and 1 by truncating at $(\alpha, (1 - \alpha))$ for some small $\alpha > 0$. In the reported simulations we used $\alpha = 0.005$. We also obtained results for $\alpha = 0.001$ or $\alpha = 0.01$, but no notable difference was observed.

In our simulations, we also included the A-IPTW estimator of ψ_0, defined as

$$\psi_n^{A-IPTW} = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{2A_i - 1}{g_n(A_i \mid W_i)}(Y_i - \bar{Q}_n^0(A_i, W_i)) + (\bar{Q}_n^0(1, W_i) - \bar{Q}_n^0(0, W_i))\right\}.$$

The two TMLEs and the A-IPTW estimator are double robust so that these estimators will be consistent for ψ_0 if either g_n or $\bar{Q}_n^0$ is consistent for g_0 and $\bar{Q}_0$, respectively. In addition, the two TMLEs and the A-IPTW estimator are asymptotically efficient if both g_n and $\bar{Q}_n^0$ consistently estimate the true g_0 and $\bar{Q}_0$, respectively.

In this simulation study we will use simple parametric maximum likelihood estimators as initial estimators $\bar{Q}_n^0$ and g_n, even though we recommend the use of super learning in practice. The goal of this simulation is to investigate the performance of the updating step under misspecified and correctly specified $\bar{Q}_n^0$, and for that purpose we can work with parametric maximum likelihood estimation fits.

We also report the MLE $\Psi(Q_n^0)$ of ψ_0 according to a parametric model for $\bar{Q}_0$, and an IPTW estimator of ψ_0 that uses g_n as estimator of g_0:

$$\psi_n^{MLE} = \frac{1}{n}\sum_{i-1}^{n}\left\{\bar{Q}_n^0(1, W_i) - \bar{Q}_n^0(0, W_i)\right\},$$

$$\psi_n^{IPTW} = \frac{1}{n}\sum_{i=1}^{n}(2A - 1)\frac{Y_i}{g_n(A_i, W_i)}.$$

The MLE of ψ_0 is included for the sake of evaluating the bias reduction step carried out by the TMLEs and the A-IPTW estimator.

7.3.2 Data-Generating Distributions

Covariates W_1, W_2, W_3 were generated as independent binary random variables: $W_1, W_2, W_3 \sim Bernoulli(0.5)$. Two treatment mechanisms were defined that differ only in the values of the coefficients for each covariate. They are of the form

$$g_0(1 \mid W) = \text{expit}(\beta W_1 + \delta W_2 + \gamma W_3).$$

We considered the following two settings for the treatment mechanism:

$$\beta_1 = 0.5, \delta_1 = 1.5, \gamma_1 = -1, \text{ and}$$
$$\beta_2 = 1.5, \delta_2 = 4.5, \gamma_2 = -3.$$

We refer to these two treatment mechanisms as $g_{0,1}$ and $g_{0,2}$, respectively. The observed outcome Y was generated as

$$Y = A + 2W_1 + 3W_2 - 4W_3 + e,\ e \sim N(0, 1).$$

For both simulations the true additive causal effect equals one: $\psi_0 = 1$. Treatment assignment probabilities based on mechanism $g_{0,1}$ range from 0.269 to 0.881, indicating no sparsity in the data for simulation 1. In contrast, treatment assignment probabilities based on mechanism $g_{0,2}$ range from (0.047 to 0.998). Simulation 2 poses a more challenging estimation problem in the context of sparse data.

Estimates were obtained for 1000 samples of size $n = 1000$ from each data-generating distribution. Treatment assignment probabilities were estimated using a correctly specified logistic regression model. In both simulations predicted values for $g_n(A \mid W)$ were bounded away from 0 and 1 by truncating at $(p, 1 - p)$, with $p = 0.01$. In one set of results a correctly specified main terms regression model was used to compute the initial estimate $\bar{Q}_n^0$, while in the other set of results the initial estimate was defined as the least squares regression Y on A only.

7.3.3 Results

Table 7.1 reports the average estimate, bias, empirical variance, and MSE for each estimator, under different specifications of the initial estimator $\bar{Q}_n^0$. In simulation 1, when $\bar{Q}_0$ is correctly estimated, all estimators perform quite well, though as expected IPTW is the least efficient. However, when $\bar{Q}_0$ is incorrectly estimated, the MLE is biased and has high variance relative to the other estimators. Since $g_n(A \mid W)$ is correctly specified, IPTW and A-IPTW provide unbiased estimates, as do both TMLEs: the TMLE_{Y^*} based on the logistic regression model is similar to the TMLE based on the linear regression model, as there is no sparsity in the data, and both are asymptotically efficient estimators.

Table 7.1 Estimator performance for simulations 1 and 2 when the initial estimator of $\bar{Q}_0$ is correctly specified and misspecified. Results are based on 1000 samples of size $n = 1000$, g_n is consistent,and bounded at (0.01,0.99)

	$\bar{Q}_0$ correctly specified				$\bar{Q}_0$ misspecified			
	ψ_n	Bias	Var	MSE	ψ_n	Bias	Var	MSE
Simulation 1								
MLE	1.003	0.003	0.005	0.005	3.075	2.075	0.030	4.336
IPTW	1.006	0.006	0.009	0.009	1.006	0.006	0.009	0.009
A-IPTW	1.003	0.003	0.005	0.005	1.005	0.005	0.010	0.010
TMLE_{Y^*}	0.993	−0.007	0.005	0.005	0.993	−0.007	0.006	0.006
TMLE	0.993	−0.007	0.005	0.005	0.993	−0.007	0.006	0.006
Simulation 2								
MLE	1.001	0.001	0.009	0.009	4.653	3.653	0.025	13.370
IPTW	1.554	0.554	0.179	0.485	1.554	0.554	0.179	0.485
A-IPTW	0.999	−0.001	0.023	0.023	1.708	0.708	0.298	0.798
TMLE_{Y^*}	0.989	−0.011	0.037	0.037	0.722	−0.278	0.214	0.291
TMLE	0.986	−0.014	0.042	0.042	−0.263	−1.263	2.581	4.173

In simulation 2, all estimators except IPTW are unbiased when $\bar{Q}_0$ is correctly estimated. In this case, both TMLEs have higher variance than A-IPTW, even though all three are asymptotically efficient. All three are more efficient than IPTW but less efficient than MLE. Though asymptotically the IPTW estimator is expected to be unbiased in this simulation, since g_n is a consistent estimator of $g_{0,2}$, these results demonstrate that in finite samples, heavily weighting a subset of observations not only increases variance but can also bias the estimate.

When the model for $\bar{Q}_0$ is misspecified in simulation 2, MLE is even more biased than it was in simulation 1. The efficiency of all three double robust efficient estimators suffers in comparison with simulation 1 as well. Nevertheless, TMLE_{Y^*}, using the logistic fluctuation, has the lowest MSE of all estimators. Its superiority over TMLE, using linear least squares regression, in terms of bias and variance is clear. TMLE_{Y^*} also outperforms A-IPTW with respect to both bias and variance and performs much better than IPTW or MLE.

7.4 Discussion

For the sake of demonstration, we considered estimation of the additive causal effect. However, the same TMLE, using the logistic fluctuation, can be used to estimate other point-treatment causal effects, including parameters of a marginal structural model. The proposed quasi-log-likelihood loss function can be used to define a super learner for prediction of a bounded continuous outcome. It will be of interest to evaluate such a super learner relative to a super learner that does not incorporate these known bounds. The quasi-log-likelihood loss function and the logistic fluctu-

ation function can also be applied in a TMLE of the causal effect of a multiple time point intervention in which the final outcome is bounded and continuous. In this case, one uses the loss function and logistic fluctuation function to fluctuate the last factor of the likelihood of the longitudinal structure. Our simulations show that the proposed fluctuation function and loss function, and corresponding TMLEs, should also be used for continuous outcomes for which no a priori bounds are known. In this case, one simply uses the minimal and maximal observed outcome values. In this way, these choices naturally robustify the TMLEs by enforcing that the updated initial estimator will not predict outcomes outside the observed range. TMLE using the logistic fluctuation function can also be incorporated in C-TMLE (Chaps. 19–21 and 23) without modification.

Appendix

The following lemma proves that the quasi-log-likelihood loss function is indeed a valid loss function for the conditional mean $\bar{Q}_0$ of a continuous outcome in $[0, 1]$.

Lemma 7.1. *We have that*

$$\bar{Q}_0 = \underset{\bar{Q}}{\operatorname{argmin}}\, E_0 L(\bar{Q}),$$

where the minimum is taken over all functions of (A, W) *that map into* $[0, 1]$. *In addition, given a function* H^**, define the fluctuation function*

$$logit(\bar{Q}(\epsilon)) = logit(\bar{Q}) + \epsilon H^*.$$

For any function H^* *we have*

$$\frac{d}{d\epsilon} L(\bar{Q}(\epsilon))\Big|_{\epsilon=0} = H^*(A, W)(Y - \bar{Q}(A, W)).$$

Proof. Let $\bar{Q}_1$ be a local minimum of $\bar{Q} \to E_0 L(\bar{Q})(O)$, and consider the fluctuation function $\epsilon \to \bar{Q}_1(\epsilon)$ defined above. Then the derivative of $\epsilon \to E_0 L(\bar{Q}_1(\epsilon))$ at $\epsilon = 0$ equals zero. However, we also have

$$-\frac{d}{d\epsilon} L(\bar{Q}_1(\epsilon))\Big|_{\epsilon=0} = H^*(A, W)(Y - \bar{Q}_1(A, W)).$$

Thus, it follows that

$$E_0[H^*(A, W)(Y - \bar{Q}_1(A, W))] = E_0[H^*(A, W)(\bar{Q}_0 - \bar{Q}_1)(A, W)].$$

But this needs to hold for any function $H^*(A, W)$, which proves that $\bar{Q}_1 = \bar{Q}_0$ almost everywhere. The final statement follows as well. □